Introduction (Ask a Question)

The Core Tri-Speed Ethernet Advanced High-performance Bus (TSE_AHB) provides 10/100/1000 Mbps Ethernet Media Access Controller (MAC) with a Gigabit Media Independent Interface (G/MII) and Ten-bit Interface (TBI) to support 1000BASE-T and 1000BASE-X.

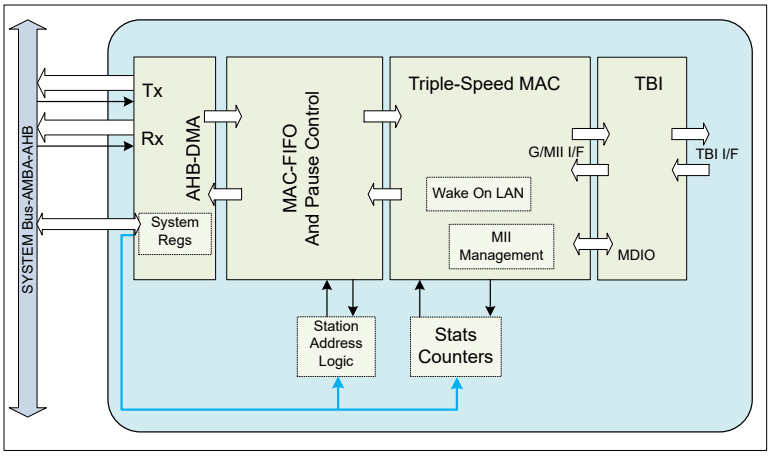
CoreTSE_AHB has the following major interfaces:

- Advanced Microcontroller Bus Architecture (AMBA®) AHB-initiator port interface allows data movement by Direct Memory Access (DMA) engine between system memory and local Tx/Rx data buffers
- G/MII or TBI Physical Layer (PHY) interface connects to Ethernet PHY
- Management Data Input/Output (MDIO) interface to communicate with the MDIO manageable device in the PHY

The primary function of the CoreTSE_AHB is delivered through a triple-speed MAC core, which encompasses features for collecting statistics and managing station addresses. This system compiles statistical data from the traffic transmitted and received across the Ethernet connection. Additionally, the Station Address Logic (SAL) feature offers the capability for address filtering.

The following figure shows the high-level block diagram.

Figure 1. CoreTSE_AHB Block Diagram



CoreTSE_AHB Summary (Ask a Question)

Core Version	This document applies to CoreTSE_AHB v4.2.
Supported Device Families	• SmartFusion® 2 • IGLOO® 2 • PolarFire® • PolarFire SoC • RTG4™ • RT PolarFire
Supported Tool Flow	Requires Libero® SoC v11.7 or later releases.
Supported Interfaces	The AHB initiator features Direct Memory Access (DMA) capabilities, while the AHB target offers the register interface functionality.
Licensing	The core supports the generation of following versions: • **Evaluation:** Evaluation version is available with obfuscated and encrypted Verilog RTL with self-destruct logic. Evaluation version supports user testbench simulation and approximately four hours of functionality on silicon. Evaluation version is not license locked and is available free. • **Obfuscated:** Obfuscated version is available with obfuscated and encrypted Verilog RTL. Obfuscated version supports user testbench simulation and supports unlimited functionality on silicon. Obfuscated version is license locked. You must purchase this license separately.
Installation Instructions	CoreTSE_AHBis installed to the **IP Catalog** of automatically through the IP Catalog update function. Alternatively, CoreTSE_AHB could be manually downloaded from the catalog. Once the IP core is installed, it is configured, generated and instantiated within the for inclusion in the project.
Device Utilization and Performance	A summary of utilization and performance information for CoreTSE_AHB is listed in the Device Utilization and Performance.

CoreTSE_AHB Change Log Information

This section provides a comprehensive overview of the newly incorporated features, beginning with the most recent release. For more information about the problems resolved, see the Resolved Issues section.

Version	What's New
v4.1	Resolved an issue where MDIO was not correctly reading the PHY address.
v4.0	Added support for RTG4™ device family.
v3.1	Added RX_SLIP functionality for PolarFire® family.
v3.0	Added support for PolarFire device family.
v2.1	Added support for IGLOO® 2 device family.

Table of Contents

1. Features (Ask a Question)

CoreTSE_AHB supports the following features:

- Tri-Speed Ethernet MAC Core
- 10/100/1000 Mbps Operation
- Full-Duplex at 10/1000 Mbps
- Half-Duplex at 10/1000 Mbps
- Standard G/MII interface
- MDIO Interface for PHY Register Access
- TBI for 1000Base-T or 1000Base-X support
- Wake on LAN (WoL) with Magic Packet Detection
- Frame Statistics Counters
- Destination Address Based Filtering

2. Functional Description (Ask a Question)

This section describes the following functional blocks of CoreTSE_AHB:

- Triple speed MAC
- MAC FIFO
- Ten-bit Interface

2.1. Triple Speed MAC (Ask a Question)

This core is a full-featured 10/100/1000 Mbps MAC with standard G/MII. The MAC has built in G/MII to TBI converter, which supports 1000 Mbps with TBI. The core is capable of full-duplex operation at 10, 100, or 1000 Mbps and of half duplex operation at 10 or 100 Mbps.

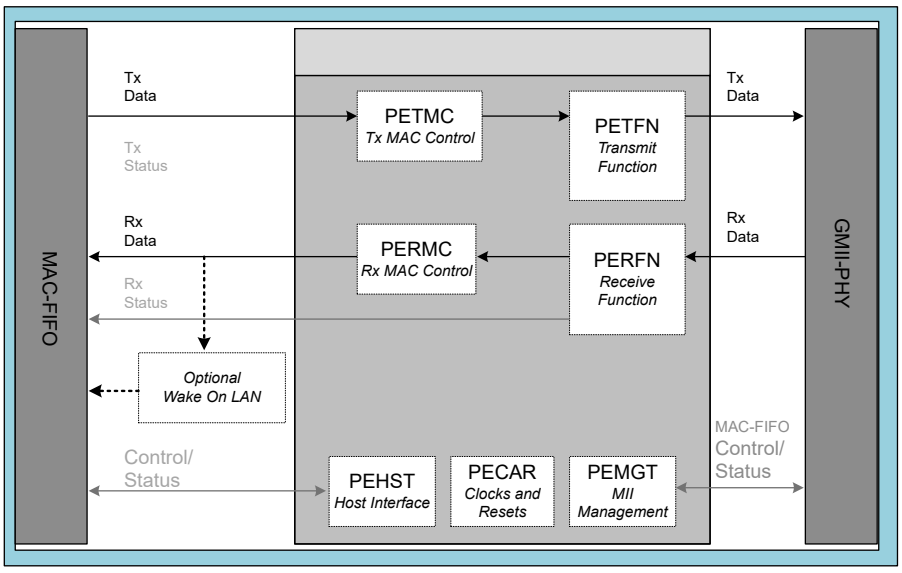
In Half-Duplex mode, the MAC compiles to the Carrier Sense Multiple Access/Collision Detect Access method as defined in IEEE® 802.3 and its several supplements including IEEE 802.3u. In Full-Duplex mode, the MAC follows IEEE 802.3x, which ignores both carrier and collisions. After each packet transmission or abortion, a transmit statistics vector is used for statistics collection.

The external PHY device presents packets to the MAC. The MAC scans the preamble searching for the Start Frame Delimiter (SFD). When the SFD is found, the preamble and SFD are stripped and the frame is passed to the system. After each frame reception, a Receive Statistics Vector is used for frame filtering and statistics collection.

CoreTSE_AHB supports PAUSE control frames and also includes optional support for Wake-On-Local-Area-Network module. The Wake on LAN (WoL) module detects both IEEE 802.3-compliant unicast frames with a destination address that matches the station address and packets that use AMD's Magic Packet™ Detection technology. The detection functionality can be enabled or disabled.

The following figure shows the triple speed MAC functional block diagram.

Figure 2-1. Triple Speed MAC Functional Block Diagram



2.1.1. PAUSE Flow Control (Ask a Question)

MAC Transmit Logic (MACTL) provides native support for PAUSE flow control frames. PAUSE frames are control frames (frames with 0x8808 as the EtherType) with a particular DA (01-80-c2-00-00-01) and the Opcode 0x0001. The FIFO-logic automatically requests to send a PAUSE frame by pulsing Transmit Control Request (TCRQ) and providing the pause time value available on control-frame-

register (CFPT [15:0]). Once a frame is received and detected as a control frame, MAC checks for the DA and the Opcode fields. If the DA is either the reserved multicast address used by PAUSE (01-80-c2-00-00-01) or the station's unique address, and the Opcode is 0x0001, then the Control frame is considered to be a PAUSE Control frame.

When a PAUSE Control frame is received:

- The MAC Receive Logic (MACRL) module indicates the MACTL to pause the stream of data frames and allows control frames transmission to the link partner. When either a PAUSE frame with a zero-value pause time is received or the MACRL pause timer expires, MACTL is considered to be unpaused and normal data frames gets resumed.
- The pause time value is loaded into the PERMC pause timer. This pause timer is a 16-bit down counter that decrements every pause quanta (a speed-independent constant of 64 byte-times). Whenever the pause time counter is nonzero, the MAC is considered to be paused and no data frames are sent.

2.1.1.1. MAC Register Configuration for Pause Frames (Ask a Question)

To send the pause frames, the following registers must be configured.

Table 2-1. MAC Register Configuration for Pause Frames

Action Description	Configuration Register Name/ Address	Description
Enable Pause Frame Transmission	MAC Configuration #1 (0x000)	Set Bit[4] to 1 to enable the MAC to generate and send Pause frames whenever necessary.
Configure Pause Quanta	MAC-FIFO Configuration Register 1 (0x04C)	Program the bits [15:0] as per the required pause quanta value.
Configure FIFO Thresholds	MAC-FIFO Configuration Register 2 (0x050)	Set the required high and low watermark levels for the receive FIFO.
Configure Full-duplex/Half-Duplex mode	MAC-FIFO Configuration Register 5 (0x05C)	Set bit [22] to '0' to operate in full-duplex mode (by default, its set to 0).
Configure Pause Timer Value	Control Frame Parameter (0x018)	Program the bit[15:0] with the desired Pause timer value.
Configure Source Address (SA)	Station Address Lower (0x040) Station Address Upper (0x044)	Program these two registers with the Source Address.

2.1.2. Jumbo Frame Support (Ask a Question)

The CoreTSE_AHB supports jumbo frames, which are frames that exceed the 1500-byte maximum of standard Ethernet frames. The CoreTSE_AHB Tx/Rx buffers can be configured in either store-and-forward or cut-through mode. In the store-and-forward mode, the Tx/Rx buffer will not be read until the entire packet is stored in the buffer. In cut-through mode, the Tx/Rx buffer begins reading as soon as the number of bytes threshold configured in "A-MCXFIF Configuration Register 1 (0x04C)" and "A-MCXFIF CONFIGURATION REGISTER 3 (0x054)" are available in the Rx and Tx buffers, respectively. The following parameter or register configurations are recommended.

Parameters Configuration: CoreTSE_AHB Tx/Rx buffers can operate in either store-and-forward mode or cut-through mode. When using store-and-forward mode, the Packet Size parameter must be set atleast twice the maximum jumbo frame size. For example, if the maximum jumbo frame size is 9K bytes, the Packet Size should be configured to 32K bytes in the CoreTSE_AHB GUI configurator. In cut-through mode, the Packet Size parameter can be set to any value.

Registers Configuration: The following registers configuration are recommended when the jumbo frame size exceeds 1500 bytes, based on the configured parameters.

1. For CoreTSE_AHB Tx/Rx buffers in store-and-forward mode, follow these steps:
   - To configure CoreTSE_AHB Rx buffer, set the **SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH)** in the **A-MCXFIF Configuration Register 1 (0x04C)** to the maximum jumbo frame size.

- To configure CoreTSE_AHB Tx buffer, set the **FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH)** in the **A-MCXFIF CONFIGURATION REGISTER 3 (0x054)** to the maximum jumbo frame size.
- To enable huge frames, set the **HUGE FRAME ENABLE** in the MAC Configuration #2 (0x004) to 1.

2. For CoreTSE_AHB Tx/Rx buffers in cut-through mode, follow these steps:
   - To configure CoreTSE_AHB Rx buffer, set the **SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH)** in the **A-MCXFIF Configuration Register 1 (0x04C)** to a value greater than 4 and less than the Packet Size parameter. It is recommended to set this to 64 bytes.
   - To configure CoreTSE_AHB Tx buffer in cut through mode, **FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH)** of the **A-MCXFIF CONFIGURATION REGISTER 3 (0x054)** to a value greater than 4 and less than half of the Packet Size parameter. It is recommended to set this to 64 bytes.
   - To enable huge frames, set the **HUGE FRAME ENABLE** in the **MAC Configuration #2 (0x004)** to 1.

2.1.3. Inter-Frame-Gap (Ask a Question)

MACRL provides the capability to filter frames that have less than a certain inter-frame gap. The standard states that the inter-frame-gap must be 160-bit times. This includes 96 bits of Inter Packet Gap (IPG), 56 bits of preamble and 8 bits of Start Frame Delimiter (SFD). To protect downstream logic from over-running, MACRL can be programmed with a minimum InterFrame Gap (IFG) parameter. The second of two back-to-back frames to violate the minimum IFG is dropped.

2.1.4. Address Detect (Ask a Question)

MACRL scans the frame to detect certain characteristics. The DA is examined to determine its address type. The 48-bit programmed station address is compared to each receive frame's DA. When they match, the UniCast Address Detect (UCAD) output is asserted. If the broadcast address is detected, MACRL asserts BroadCast Address Detect (BCAD). If a multicast address is detected, the MAC asserts MultiCast Address Detect (MCAD).

2.1.5. Hash Table Support (Ask a Question)

MACRL supports hash table with up to 128 entries. Seven bits of the Cyclic Redundancy Check (CRC) of the DA are used as the Hash Value (HASHV [6:0]).

2.1.6. Length Checking and Maximum Length Enforcement (Ask a Question)

MACRL optionally compares the length field with the actual length of the data field portion of the frame. This is enabled through the MAC Configuration #2 register. MACRL first determines if the length/type field has a valid length. If the klength is not valid, it is compared with the data field length and any mismatches are reported through the receive statistics.

MACRL limits the length of the receive frames passed to the system. The maximum length is programmed through the Maximum Frame Length register. Frames which exceed this maximum length are truncated.

2.1.7. Internal Loopback at G/MII (Ask a Question)

Asserting the internal loopback enable bit in MAC Configuration #1 register, enables the MAC transmit output's looped back to the MAC receive inputs at G/MII interface.

2.1.8. Wake on Local Area Network (WoL) (Ask a Question)

The MAC WoL is based on AMD's Magic Packet Detection technology.

The first step of the detection procedure is to scan the initial 12 bytes of the frame which contain Destination and Station addresses. Magic Packet detection is only carried out when the incoming

frame's destination address matches the MAC's station address, or if the frame's destination address is a multicast or broadcast address.

After the initial twelve bytes of the frame have matched, Core searches for the Magic Packet technology's defined preamble of six continuous aligned bytes with all bits asserted (0xFFh). Following a valid Magic Packet preamble, Core immediately expects 16 back-to-back repetitions of the six-byte MAC station address. Failure to achieve this exact pattern by a single byte at any time during the frame resets the circuitry back to the preamble search state.

After successful recognition of the Magic Packet payload or a successful compare of the MAC's station address with the incoming frame's destination address, the Interface Status register (bit field WakeOnLaneDetected) is asserted and status bit can only be cleared through assertion of the WakeOnLaneDetectedClear bit field of Interface Control register.

2.1.9. MDIO Management (Ask a Question)

Control and status is provided to and from the PHY through the two-wire MDIO management interface described in *IEEE802.3u Clause22*.

The MDIO write/read cycles are performed through the APB target. MAC performs a write cycle using the MDIO_PHYID, register address, and 16-bit write data. The MAC performs a read cycle using the MDIO_PHYID, register address, and updates the 16-bit read data into the MDIO Management Status register which is read through APB target.

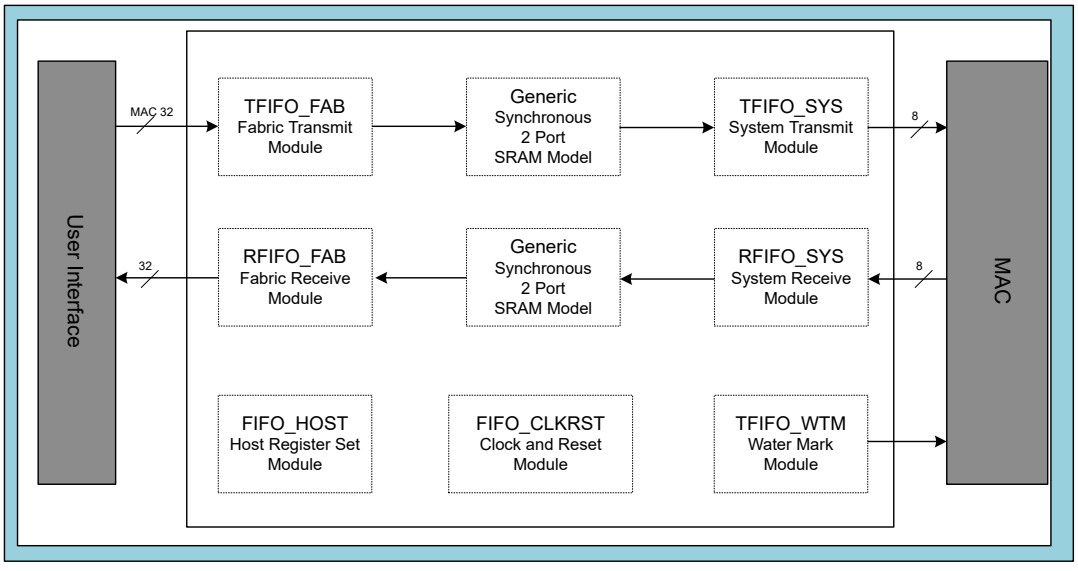
2.2. MAC FIFO (Ask a Question)

This core provides data queuing for increased throughput and integrated between back-end user-interface logic and MAC core. The core provides clock-domain crossing, automatic pause frame handshaking, and graceful frame dropping.

The data is buffered between the system-interface and the MAC core by transmit and receive FIFOs. The FIFO size is configured with PACKET_SIZE parameter.

The following figure shows MAC-FIFO functional block diagram.

Figure 2-2. MAC-FIFO Functional Block Diagram



Each RAM has additional associated control bits which are additional to max frame data size.

The following table lists the MAC-FIFO RAM configurations.

Table 2-2. MAC-FIFO RAM Configurations

PACKET_SIZE Parameter (Bytes)	Transmit RAM		Receive RAM	
	RAM Size in Bits	Number of Address bits (TABITS)	RAM Size in Bits	Number of Address bits (RABITS)
256	64x39	6	128x36	7
512	128x39	7	256x36	8
1K	256x39	8	512x36	9
2K	512x39	9	1Kx36	10
4K	1Kx39	10	2Kx36	11
8K	2Kx39	11	4Kx36	12
16K	4Kx39	12	8Kx36	13
32K	8Kx39	13	16Kx36	14

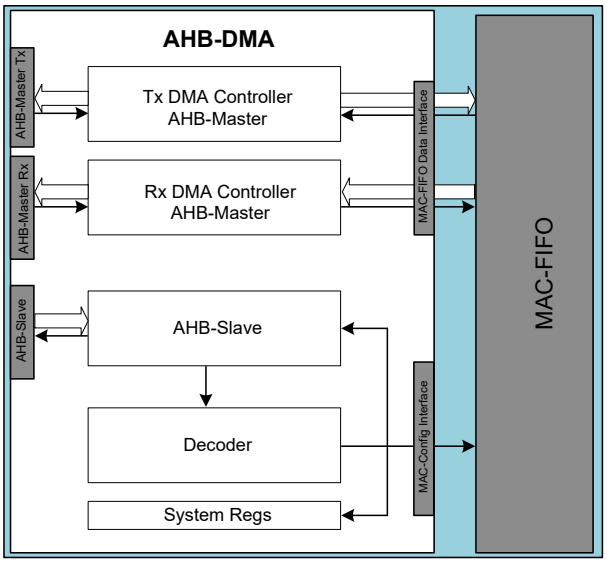
2.2.1. AMBA-AHB Compliant DMA Engines (Ask a Question)

This module provides a DMA bridge between a host-to-system that uses an AMBA AHB™ bus and the Ethernet MAC and MAC-FIFO. It interfaces to the host-system through 32-bit AHB initiator and target ports. On the MAC side of the module, it has a high-performance synchronous interface for DMA data transfer to and from the FIFO.

For ease of handling the software, transfers are handled using linked lists of transfer descriptors which together define one buffer in host memory for Tx operations and another for Rx operations. These buffers are typically configured as ring buffers, but this is up to the user to implement. Registers within the AHB-DMA provide control and status information regarding these transfers. These registers are accessed through the AHB target port, alongside accesses to the AHB Target interfaces on the MAC and the FIFO.

The following figure shows AHB-DMA functional block diagram.

Figure 2-3. AHB-DMA Functional Block Diagram



DMA always performs 32 bits operation. HSIZE port of both the TX and RX AHB Initiator is tied to 2'b10 that is, 32 bits operation only.

Because AHB does not support the data strobe feature, valid bytes must always be started from the LSB for both TX and RX.

TX: In the TX path user can pad invalid bytes with any value. DMA reads 32 bits (4 bytes in last transaction) but DMA controller sends only the valid bytes to MAC as per the configuration of packet size descriptor.

RX: In the RX path, DMA controller updates the packet size descriptor as per the number of bytes received from the MAC. In the last transfer, DMA controller writes 32 bits (including invalid bytes padded with zero if required). User must first read the packet size descriptor and must only reads the number of valid bytes written in the packet size. For the last transfers, the user can read all the 32 bits data and discards the invalid bytes.

2.2.2. Interrupt Coalescing (Ask a Question)

The Scatter-Gather feature improves the DMA operational throughput by removing the system CPU from the real-time control of the DMA. However, this improved transfer throughput can overwhelm the CPU with transfer completion interrupts. Providing a programmable filter method called interrupt coalescing solves this issue. The DMA interrupt coalescing feature is added beyond the traditional individual interrupt for each successful packet transfer. This can be enabled by setting corresponding Mask bit in the Interrupt Mask register. The default value of the interrupt threshold count is one, and this must be updated as per the application requirements.

While DMA is operational, the TX/RX PktCnt is maintained. With each local interrupt on completion of event (TXPkt transmitted/RXPkt received), PktCnt is incremented. When the count reaches the coalescing threshold value coalescing interrupt bit is set in interrupt register. This interrupt remains asserted as long as the TX/RX-Count value in the DMA status register is nonzero. To acknowledge this coalescing interrupt, software must decrement PktCnt in the DMA Status register by writing 1 to the corresponding bit after acknowledging the event (TXPkt transmitted/RXPkt received).

2.2.3. Frame Filtering (Ask a Question)

This section provides the information about different types of frame filtering.

2.2.3.1. Station Address Logic for Frame Filtering (Ask a Question)

This module provides a mechanism to statistically filter frames not intended for that node.

The MAC core performs DA comparison on all the received frames and provides three information signals: UCAD (Perfect DA match), MCAD, and BCAD along with seven most significant bits of the resulting CRC of DA. This information is used to perform a hashing algorithm, compares the result with a programmable hash table and then communicate to the FIFO logic to either delete or store the frame.

The programmability allows the user to assert user choice bits in a 128-bit hash table that correspond to the desired Ethernet MAC DA. If the corresponding bit in the table is set, the frame is accepted. In addition, hashing can selectively be performed on unicast addresses or multicast addresses.

2.2.3.2. Required Registers for Unicast Frame Filtering (Ask a Question)

To enable filtering of unicast frames based on the MAC's Destination Address (DA), the following registers must be configured:

Station Address Registers (0x040 and 0x044)

These registers store the Station Address (the expected Destination Address).

- 0x040 – Station Address Lower Register
- 0x044 – Station Address Upper Register

Example: If the required DA is **01-80-C2-00-00-01**, the registers are programmed as follows:

0x040 – Station Address Lower Register

Bit [31:24] - 8'h01 (D0)

Bit [23:16] - 8'h00 (D1)

Bit [15:8] - 8'h00 (D2)

Bit [7:0] - 8'hC2 (D3)

0x044 – Station Address Upper Register

Bit [31:24] - 8'h80 (D4)

Bit [23:16] - 8'h01 (D5)

Bit [15:0] - 16'h0000 (reserved)

Important: Bit[17] of register 0x058 (MAC-FIFO Configuration Register 4) and Bit[2] of register 0x1C0 (Frame Filter Controls) are mutually exclusive. So, only one of the following two configurations can be used at a time.

Configuration 1: Configure the two registers 0x058 and 0x05C as follows:

MAC-FIFO Configuration Register 4 (0x058)

- Bit[17] – Host filter frames
  - Setting this bit drops the frames where DA does not match the programmed Station Address.
  - Clearing this bit passes all frames even when DA does not match the programmed Station Address (no filtering).

MAC-FIFO Configuration Register 5 (0x05C)

- Bit[17] – Mask bit for unicast filtering control
  - Default is 1 which means the Bit[17] of 0x058 is masked (filtering disabled).
  - Set this bit to 0 to make the Bit[17] of 0x058 effective for the unicast filtering.

Configuration 2:

Important: This configuration will be effective only when the "Include Station Address Filtering Logic" is enabled in the configurator.

Frame Filter Controls (0x1c0)

- Bit[3] must be set to 0 (By default, this bit is set to 1)
- Bit[2] of the register must be 1. (By default, this bit is set to 1).

2.2.3.3. Required Registers for Multicast and Broadcast Frame Filtering (Ask a Question)

To enable filtering of multicast and broadcast frames, the following registers must be configured.

Important: Bit[9:8] of register 0x058 (MAC-FIFO Configuration Register 4) and Bit[1:0] of register 0x1C0 (Frame Filter Controls) are mutually exclusive. So, only one of the following configurations can be used at a time.

Configuration 1: Configure the two registers 0x058 and 0x05C as follows:

MAC-FIFO Configuration Register 4 (0x058)

- Bit[9:8] – Host filter frames
  - Bit[9]: Setting this bit drops the frames that contain broadcast address.

- Bit[8]: Setting this bit drops the frames that contain multicast address.
- Clearing these two bits passes all the broadcast and multicast frames without filtering.

MAC-FIFO Configuration Register 5 (0x05C)

- Bit[9:8] – Mask bits for multicast and broadcast filtering control
  - Default is 2'b11 which means the Bit[9:8] of 0x058 is masked (broadcast and multicast filtering is disabled).
  - Set these bits to 2'b00 to make the Bit[9:8] of 0x058 effective for the broadcast and multicast filtering respectively.

Configuration 2:

Important: This configuration will be effective only when the "Include Station address filtering logic" is enabled in the configurator.

Frame Filter Controls (0x1c0)

- Bit[3] must be set to 0 (By default, this bit is set to 1)
- Bit[1] of the register must be 1 to enable multicast frame filtering (By default, this bit is set to 1).

Bit[0] of the register must be 1 to enable broadcast frame filtering (By default, this bit is set to 1).

2.2.3.4. Required Registers for Filtering of Different Types of Error Frames (Ask a Question)

To enable filtering of error frames received, the following registers must be configured. In addition to unicast, multicast, and broadcast filtering, bits [17:0] of 0x058 and 0x05C registers filter different types of error frames.

- To enable filtering for a specific frame type, set the corresponding bit in register 0x058.
- To make this setting effective, clear the matching bit in register 0x05C (as the 0x05C acts as the mask for 0x058).

2.2.4. Statistics Counters Logic (Ask a Question)

This module has separate counters, which counts or accumulates conditions that occur when the packets are transmitted or received. These counters support Remote Network Monitoring (RMON) management information base (MIB) group 1, RMON MIB group 2, RMON MIB group 3, RMON MIB group 9, RMON MIB 2, and the dot 3 Ethernet MIB.

2.2.4.1. Recommended Sequence to Clear CARRY Registers CAR1 and CAR2 Along With Stats Counters (Ask a Question)

It is always recommended to clear the Stats counter register first, followed by clearing the respective carry bit of CAR1/CAR2.

In the user guide, it was mentioned that CAR1 and CAR2 registers are of Read or Write-1-to-Clear (R/W1C) type.

The following points summarize the behavior and recommended usage of the CAR1 and CAR2 registers:

- When reading these registers, each bit indicates the status of a respective **Stats Counter Carry Bit** (or "carry flag").
  Note: If any bit in CAR1/CAR2 is high, the Stats Carry Interrupt (TSM_INTR[1]) is asserted.
- Writing '1' into a specific bit in CAR1/CAR2 clears the corresponding carry flag and also deasserts the TSM_INTR[1] interrupt (if no other carry flags are set).
- Writing '1' to CAR1/CAR2 only clears the carry flag/interrupt, it does not clear the actual **Stats Counter** data. To avoid ambiguity and ensure that the subsequent reads of the stats counter

are accurate, it is always recommended to clear the **Stats Counter Register** first, followed by clearing the respective **Carry Bit**.

- To clear the stats counter, perform either of the following steps:
  - Configure the Bit[4]: "STATS COUNTERS: AUTO CLEAR COUNTERS ON READ" of the **Interface Control (0x038)** register. This is a one-time configuration and any stats counter data will be cleared upon read.
  - Set the Bit[3]: "STATS COUNTERS: CLEAR ALL COUNTERS" of the **Interface Control (0x038)** register to 1 and this clears "all the stats counters" data at once.

2.2.5. COMMA Alignment Logic (Ask a Question)

The PHY layer includes COMMA alignment logic in the receive path. This logic detects COMMA data and aligns the 10-bit data to the proper word boundary before passing the data to the receive path.

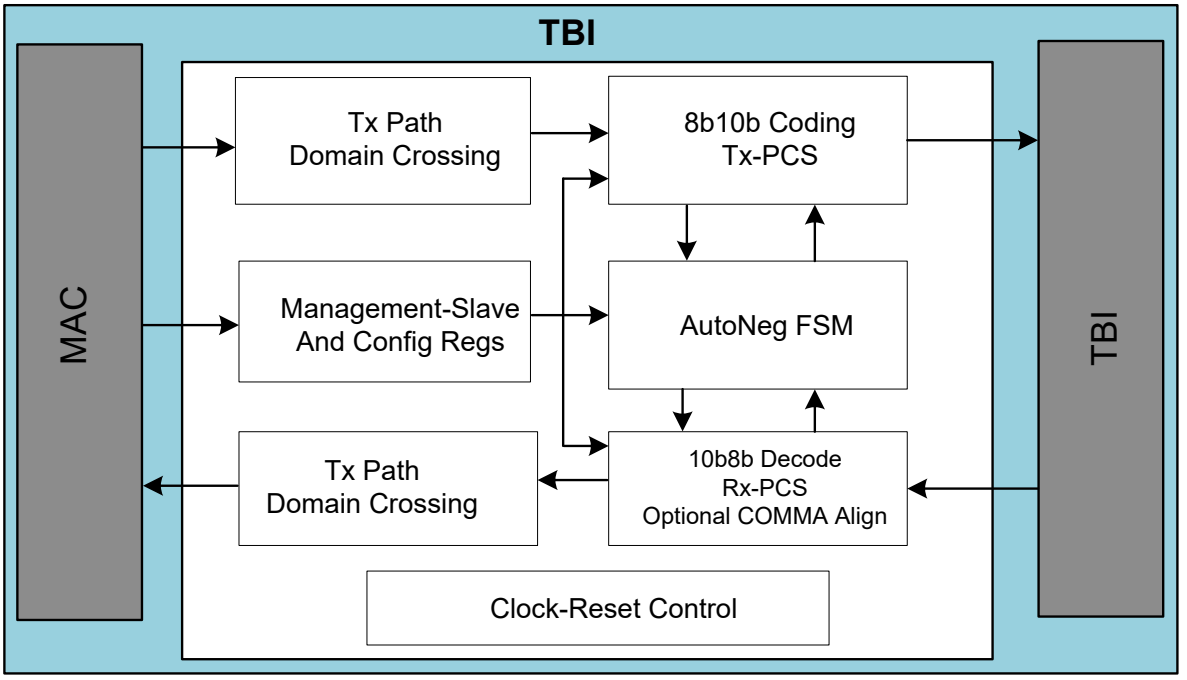
2.3. Ten-Bit Interface (Ask a Question)

This module takes the transmit G/MII data stream, encodes it into 10-bit symbols and presents 10-bit interface data to SerDes. Packet data replication is used to match data rates for the different modes of the MII to the transmit clock. In the receive direction de-serialized 10-bit symbols are decoded and converted into the receive G/MII signal set. Packet data under sampling is used to match data rates for the different modes of the MII to the TBI receive clock.

The design uses transmit, receive, and synchronization state machines as specified in IEEE 802.3z Clause 36. Also included Auto-Negotiation (AN) for 1000BASE-X, which is used to exchange information between the link partners. This module is managed and monitored through the MDIO management interface. The extended set of management registers is provided.

The following figure shows the TBI functional block diagram.

Figure 2-4. TBI Functional Block Diagram



Both the transmit and receive paths leverage the physical coding sublayer and the auto-negotiation sub-layers of the IEEE 802.3z specification, as contained in Clauses 36 and 37. For complete clock domain isolation of the TBI from the MAC, both transmit and receive elasticity FIFOs are used.

The control information exchanged differs from the IEEE specification. Instead of using the ability advertisement, the PHY sends the control information through its `Tx_config_Reg[15:0]`, as listed in Table 2-3. After receiving control information, the MAC asserts the bit 14 of its `Tx_config_Reg[15:0]` and acknowledges the update of the control information.

To maintain a constant clock frequency at the PHY interface for all MAC speeds, the MII bus data must be replicated internally to the TBI. Nibble packet data transmitted by a 100 Mbps MAC must be aligned, concatenated, and replicated 10 times. Nibble packet data transmitted by a 10 Mbps MAC must be aligned, concatenated, and replicated 100 times.

Table 2-3. TBI Auto-Negotiation Control Information Sent/Received

Bit	Tx_config from PHY to MAC	Tx_config from MAC to PHY
15	Link: 1: Link up 0: link down	0: Reserved
14	Reserved for AN ACK	1
13	0: Reserved	0: Reserved
12	Duplex mode: 1: Full 0: Half	0: Reserved
11:10	Speed: Bit[11:10] 11: Reserved 10: 1000 Mbps 01: 100 Mbps 00: 10 Mbps	0: Reserved
9:1	0: Reserved	0: Reserved
0	1	1

Before sending to a 100 Mbps MAC, paacket data received by the TBI through the PHY must be undersampled by a factor of 10. Before sending to a 10 Mbps MAC, packet data received by the TBI through the PHY must be undersampled by a factor of 100. For half-duplex functionality, carrier sense is inferred from RXDV, and collision is derived from the simultaneous assertion of TXEN and RXDV.

3. Programmer Guide (Ask a Question)

This section has all the information regarding usage of the core.

3.1. Functional Overview (Ask a Question)

The AHB-DMA Bridge core includes two-channel DMA-Controller for transmiting and receiving the data path operations. The transfer of data in either direction typically uses a ring buffer defined within host memory. The ring buffers for transmit and receive operations are defined by a closed linked list of Tx/Rx descriptors. The two ring buffers are formed of equal-sized segments, each of which is 32-bit aligned and is capable of storing a packet of up to the maximum size of packet transferred.

The way transmit operations and receive operations are carried is as described in Table 3-1. The requirement of the AHB specification that burst transfers must not cross 1Kbyte boundaries is handled seamlessly by the AHB-DMA Bridge core, and does not affect the operation of the Ethernet components of the system. Once initialized, the DMA controller can be left unattended to continuously fill/empty the specified ring buffers. Software either uses the DMA interrupts generated or polls semaphore bits within the descriptors, to maintain synchronization with the packet streams.

3.1.1. Descriptor Information (Ask a Question)

Before any DMA transfers can be carried out, two sets of descriptors need to be initialized in the host memory, one for transmit operations and the other for receive operations. Each set of descriptors takes the form of a linked list (typically closed to form a ring buffer).

The following table lists DMATX/RX descriptor information.

Table 3-1. DMATX/RX Descriptor Information

Address	Register	Function	Size
+0h	PacketStartAddr	Start address for the packet data	32 bits
+4h	PacketSize	Size of packet, Overrides and Empty Flag	32 bits
+8h	NextDescriptor	Location of next descriptor	32 bits

The entry point into the buffer used at the start of any sequence of transfers is given by the Descriptor picked out by the DMATx/RxDescriptor register. Each descriptor comprises a sequence of three 32-bit memory locations. The following table lists the AHB-DMA descriptor information.

Table 3-2. AHB-DMA Descriptor Information

Address Offset	Function
0x000	Start address for packet data [31:2] (R/W) Top 30 bits of Packet Start Address. Default 0x00 The built-in DMA controller reads this register to discover the location in host memory of the first byte of data. [1:0] (R/W) Ignored; Default is 0. Ignored by the DMA controller, because it is a requirement of the system that all transfers are 32-bit aligned in the host memory.

Table 3-2. AHB-DMA Descriptor Information (continued)

Address Offset	Function
0x004	Packet Information [31] (R/W) Empty Flag For transmit operations, this bit indicates the availability of the data associated with the packet. For receive operations, this bit indicates the availability of the specified location to store the received packet. The setting of this flag is used to validate the descriptor. **Note:** On successful completion of a transmit operation, the DMA controller writes 1 to this location to indicate that the associated data has been transferred from this location. On successful completion of a receive operation, the DMA controller writes 0 to this location to indicate that this location has been used to store the received packet. The first action ensures that the data cannot be accidentally transferred twice, the second action ensures that received data is not accidentally overwritten by a subsequent packet. [30:21] Reserved [20:16] (R/W) FTPP Overrides 5-bit field containing the per-packet override flags signaled to the FIFO during packet transmission. The following are the encoded bits: • 20: FIFO Transmit Control Frame • [19:18]: FIFO Transmit Per-Packet PAD Mode • 17: FIFO Transmit Per-Packet Generate FCS • 16: FIFO Transmit Per-Packet Enable [15:0] (R/W) Packet Size 16-bit field which, for transmit operations, gives the size of packet to be transferred in bytes. In receive operations, the DMA controller writes the number of bytes received to this field: the value of this field prior to the transfer being made is ignored.
0x008	Pointer to Next Descriptor [31:2] (R/W) Top 30 bits of Descriptor Address The built-in DMA controller reads this register to discover the location in host memory of the descriptor for the next packet in the sequence. The descriptors must be from a closed linked list. [1:0] (R/W) Ignored; Default 0x0 Ignored by the DMA controller, because it is a requirement of the system that all descriptors are 32-bit aligned in host memory

3.1.2. Transmit Operation (Ask a Question)

Before any packets are transmitted, a group of Tx descriptors needs to be setup to define the ring buffer used for transmit operations. The start addresses set for the different segments of the ring buffer are required to be 32-bit aligned and must be spaced to give segments of equal size, each able to handle a packet of the maximum size to be transferred. Additionally, the PacketSize elements within these descriptors must be initially configured with the value of 1 in bit 31 (designated as the Empty Flag) to indicate that the ring buffer does not currently contain any valid data.

The bottom four bits of the DMAIntrMask register also need to be set to specify which Tx DMA events will cause a DMA interrupt to be generated. The data for one or more transmit packets must be placed in contiguous segments of the ring buffer, and the PacketSize component of the descriptor associated with these segments amended both to record the size of the packet placed in the buffer and to set the Empty Flag to 0, to indicate the presence of valid data. (Further packets for transmission is written to the ring buffer as required, as long as segments are available within the ring buffer to accommodate these packets. Available segments are indicated by a 1 in bit 31 of the PacketSize component of the associated Tx descriptor.)

The location of the descriptor corresponding to the required entry point into the Tx ring buffer must be written to the DMATxDescriptor register and the TxEnable bit (bit 0 of the DMATxCtrl register) set to enable DMA transfer of transmit packets.

The built-in DMA controller then reads DMATxDescriptor to discover the location of the first Tx descriptor, then read that descriptor initially to check the validity of the associated packet (indicated by the Empty Flag in bit 31 of the PacketSize component of the descriptor being set to 0) then to discover the start address of the packet to be transmitted and its size. (If the Empty Flag is 1, the descriptor is not currently associated with valid data. Where this is the case, the DMA controller terminates the sequence of transmit packet transfers, set the TxUnderrun bit in the DMATxStatus register and clear the TxEnable bit in the DMATxCtrl register. If enabled, an interrupt is generated with the DMAInterrupts register showing TxUnderrun as the source of this interrupt. Any further transfers require the DMATxDescriptor register to be updated to record the start position in the ring buffer that is now required and to set the TxEnable bit to 1 again.)

Once the transfer is successfully completed, the DMA controller writes 1 to bit-31 of the PacketSize component of the descriptor. The TxPktSent flag in the DMATxStatus register is also set (if not already set), the TxPktSent interrupt is generated (if enabled) and the TxPktCount recorded in bits [23:16] of the DMATxStatus register is incremented by 1.

The DMA controller then moves on to process any packet stored in the next segment of the ring buffer. The location of the descriptor associated with the next segment in the ring is already read from the NextDescriptor component of the current descriptor.

If a bus error occurs, the DMA controller terminates the sequence of transmit packet transfers, set the Bus Error bit in the DMATxStatus register and clear the TxEnable bit in the DMATxCtrl register. If enabled, an interrupt is generated with the DMAInterrupts register showing a Tx Bus Error as the source of this interrupt. Any further transfers require the DMATxDescriptor register to be updated to record the new start position in the ring buffer and the TxEnable bit to be set to 1 again.

3.1.3. Receive Operation (Ask a Question)

To receive packets, Rx descriptors must be configured to define the ring buffer for receive operations. The start addresses for the different segments of the ring buffer must be 32-bit aligned and evenly spaced to handle maximum packet sizes. The PacketSize components of these descriptors should have bit 31 set to 1 (the Empty Flag) to indicate that the ring buffer currently does not contain any received packets. Additionally, bits[7:4] of the DMAIntrMask register must be set to specify which Rx DMA events triggers a DMA interrupt. Once these configurations are in place, the descriptor's location for the entry point into the Rx ring buffer must be written to the DMARxDescriptor register, and the RxEnable bit (bit 0 of the DMARxCtrl register) must be set to enable DMA transfer of received packets.

The built-in DMA controller reads the DMARxDescriptor to find the location of the first Rx descriptor. It then checks this descriptor to ensure that the associated host memory area is available for storing the received packet (indicated by the Empty Flag in bit 31 of the PacketSize component of the descriptor being set to 1). The controller also discovers the start address of this storage area from the descriptor.

If the Empty Flag is 0, it indicates that the storage area already contains a packet that has not been read by the host software. In this case, the DMA controller terminates the sequence of Receive packet transfers, sets the RxOverflow bit in the DMARxStatus register, and clears the RxEnable bit in the DMARxCtrl register. If interrupts are enabled, an interrupt is generated with the DMAInterrupts register indicating RxOverflow as the source. To resume transfers, the DMARxDescriptor register mus be updated to the new start position in the ring buffer, and the RxEnable bit must be set to 1 again.

If the transfer is successfully completed, the DMA controller records the number of bytes transferred in bits[11:0] of the PacketSize component of the descriptor and sets bit 31 to 0 to indicate that a packet has been stored in the ring buffer. The RxPktReceived flag in the DMARxStatus register is set (if not already set), a RxPktReceived interrupt is generated (if enabled), and the RxPktCount recorded in bits[23:16] of that register is incremented by 1. If Rx FIFO ready signal continues to be asserted, the DMA controller moves on to transfer the next packet to the next segment of the ring buffer, using the location of the descriptor from the NextDescriptor component

of the current descriptor. The software must respond to the RxPktReceived interrupt by reading the packet from its location in the ring buffer and then setting the Empty Flag in the descriptor to 1 again, marking this segment of the ring buffer as available for storing further received packets.

If a bus error occurs, the DMA controller terminates the sequence of Receive packet transfers, sets the Bus Error bit in the DMARxStatus register, and clears the RxEnable bit in the DMARxCtrl register. If interrupts are enabled, an interrupt is generated with the DMAInterrupts register indicating an Rx Bus Error as the source. To resume transfers, the DMARxDescriptor register must be updated to the new start position in the ring buffer, and the RxEnable bit must be set to 1 again.

4. CoreTSE_AHB Parameters and Interface Signals (Ask a Question)

This section discusses the parameters in the CoreTSE_AHB GUI configurator and I/O signals.

4.1. Configuration GUI Parameters (Ask a Question)

The Register Transfer Level (RTL) code for CoreTSE_AHB has parameters for configuring the core. While working with the core in the tool, a configuration GUI is used to set the values of these parameters.

Important: The Parameter Name column in the following table lists the actual parameter name used in RTL. The Description column starts with the parameter names as they appear in the Core Configurator (In Bold text).

The CoreTSE_AHB parameters are listed in the following table.

Table 4-1. Parameter Descriptions

Parameter Name	Valid Range	Default	Description
GMII_TBI	0: G/MII 1: TBI	TBI	**Select Interface** 0: GMII is active 1: TBI is active This Parameter selects the interface
PACKET_SIZE	6: 256 B 7: 512 B 8: 1 KB 9: 2 KB 10: 4 KB 11: 8 KB 12: 16 KB 13: 32 KB	8K Bytes	**MAC TX/RX FIFO Depth** PACKET_SIZE parameter is used in the design to define Transmit and Receive FIFO address width. Supported PACKET_SIZE choices: 256 B 512 B 1 KB 2 KB 4 KB 8 KB 16 KB 32 KB
SAL	0 and 1	1	**Include station address frame filtering logic** 0: Do not include 1: Include
WoL	0 and 1	1	**Include Wake on LAN logic** 0: Do not include 1: Include Supports Wake on LAN using AMD's Magic Packet Detection technology Include Wake on LAN (WoL) detection logic
STATS	0 and 1	1	**Include Statistics counters logic** 0: Do not include 1: Include
MDIO_PHYID	0–31	18	**MDIO PHY Address** MDIO Physical Address, it is an integer value

Table 4-1. Parameter Descriptions (continued)

Parameter Name	Valid Range	Default	Description
SLIP_ENABLE	0 and 1	0	**Include receive slip logic** 0: Do not include 1: Include **Notes:** • Applicable for PolarFire® only • SLIP_ENABLE has to be configured according to Transceiver's SLIP configuration
ECC_ENABLE	0 and 1	0	**ECC Enable** 0: ECC is disabled for embedded memories used in transmit and receive packet FIFO 1: ECC is enabled for embedded memories used in transmit and receive packet FIFO **Note:** When this parameter is disabled, Interrupts related to ECC in the interrupt ports is '0' always
TXRX_INTR_ENABLE	0 and 1	1	**Enable Separate Tx and Rx Interrupt ports** 0: Enables common Interrupt port "TSM_INTR" for both Tx and Rx interrupts 1: Enables separate Interrupt ports "TSM_TX_INTR" for Tx interrupts and "TSM_RX_INTR" for Rx interrupts

4.2. Input and Output Signals (Ask a Question)

The ports present on CoreTSE_AHB are listed in the following tables.

Table 4-2. Clock Ports

Port Name	Width	Direction	Description
TXCLK	1	Input	2.5/25/125 MHz transmit clock generated from Transceiver TX clock according to 10/100/1000 Mbps support
RXCLK	1	Input	2.5/25/125 MHz receive clock generated from Transceiver RX clock according to 10/100/1000 Mbps support
TBI_TX_CLK	1	Input	125 MHz TBI from transmit clock of Transceiver
TBI_RX_CLK	1	Input	125 MHz TBI from receive clock of Transceiver
HCLK	1	Input	AHB system clock, maximum 100 MHz
MDC	1	Output	MDIO management Data Clock. This clock is generated from the HCLK.

Table 4-3. Reset Ports

Port Name	Width	Direction	Clock Domain	Description
HRESETN	1	Input	HCLK	AHB SYSTEM Reset (active-low) This active-low reset resets the core to its initial state. This IP core has internal reset synchronizers per clock domain. For RTG4 family devices, HRESETN is used as a synchronous reset and to ensure successful synchronization this signal must be asserted for a minimum of three clock cycles of the slowest clock frequency. For other than RTG4™ family devices, HRESETN is used as an asynchronous reset and it is synchronized in all the clock domain such that synchronized reset get asserted asynchronously and deassert synchronously.

Table 4-4. Input and Output Ports

Port Name	Width	Direction	Clock Domain	Description
TBI/Transceiver Interface Signals				
RCG	10	Input	TBI_RX_CLK	Receive code group from Transceiver
TCG	10	Output	TBI_RX_CLK	Transmit code group to Transceiver
ANX_STATE	10	Output	TBI_TX_CLK	Auto negotiation status information 0th bit: DISABLE_LINK_OK state 1st bit: AN_ENABLE state 2nd bit: AN_RESTART state 3rd bit: ABILITY_DETECT state 4th bit: ACKNOWLEDGE_DETECT state 5th bit: NEXT_PAGE_WAIT state 6th bit: COMPLETE_ACKNOWLEDGE state 7th bit: IDLE_DETECT state 8th bit: LINK_OK state 9th bit: Received configuration frame data
SYNC	1	Output	TBI_RX_CLK	Receive link sync status
SIGNAL_DETECT	1	Input	TBI_RX_CLK	The SIGNAL_DETECT is typically provided from the optical module to indicate when an optical signal is valid, otherwise this port signal must be driven “HIGH’.
TBI_RX_READY	1	Input	TBI_TX_CLK	RCG valid, recommended to connect with Transceiver Ready

Table 4-4. Input and Output Ports (continued)

Port Name	Width	Direction	Clock Domain	Description
TBI_TX_VALID	1	Output	TB_TX_CLK	TCG valid, recommended to connect with Transceiver transmit valid **Note:** This signal is visible to the user when SmartFusion® 2/IGLOO® 2/RTG4™ families are used.
TBI_RX_VALID	1	Input	TB_RX_CLK	Available for PolarFire®, recommended to connect to Transceiver receive valid
RX_SLIP	1	Output	TB_RX_CLK	Available for PolarFire, recommended to connect to receive the slip signal of Transceiver **Note:** This port is visible to the user only when Include receive slip logic is enabled in the GUI configurator.
G/MII Interface Signals				
TXD	8	Output	TXCLK	Transmit Data TXD[3:0] used for MII 100/10 Mbps (Nibble Mode) TXD[7:0] used for GMII 1000 Mbps (Byte Mode)
TXEN	1	Output	TXCLK	Transmit Enable
TXER	1	Output	TXCLK	Transmit Error
RXD	8	Input	RXCLK	Receive Data RXD[3:0] used for MII 100/10 Mbps (Nibble Mode) RXD[7:0] used for GMII 1000Mbps (Byte Mode)
RXDV	1	Input	RXCLK	Receive Data Valid
RXER	1	Input	RXCLK	Receive Error
CRS	1	Input	Asynchronous	G/MII carrier sense flag
COL	1	Input	Asynchronous	G/MII collision detect flag
Management Interface MDIO Signals				
MDI	1	Input	HCLK	MDIO management Data Input from pad
MDO	1	Output	HCLK	MDIO management Data Output
MDOEN	1	Output	HCLK	MDIO management Data Output Enable
AHB-Target Port Interface Signals				
HWDATA	32	Input	HCLK	AHB write data
HADDR	32	Input	HCLK	AHB address
HSEL	1	Input	HCLK	AHB Target select
HTRANS	2	Input	HCLK	AHB transaction type
HWRITE	1	Input	HCLK	AHB transaction direction
HRDATA	32	Output	HCLK	AHB Target read data
HRESP	2	Output	HCLK	AHB Target response
HREADYOUT	1	Output	HCLK	AHB Target ready
HMASTLOCK	1	Input	HCLK	Locked sequence of transfers
HBURST	3	Input	HCLK	AHB Burst transfers
HSIZE	3	Input	HCLK	Size of the transfer
HREADY	1	Input	HCLK	AHB ready
Tx-DMA AHB-Initiator Interface Signals				
TXHGRANT	1	Input	HCLK	AHB bus grant
TXHRESP	2	Input	HCLK	AHB response
TXHRDATA	32	Input	HCLK	AHB read data
TXHBUSREQ	1	Output	HCLK	AHB Initiator bus request
TXHTRANS	2	Output	HCLK	AHB Initiator transaction type
TXHADDR	32	Output	HCLK	AHB Initiator transaction address

Table 4-4. Input and Output Ports (continued)

Port Name	Width	Direction	Clock Domain	Description
TXHWRITE	1	Output	HCLK	AHB Initiator transfer direction
TXHWDATA	32	Output	HCLK	AHB Initiator write data bus
TXHREADY	1	Input	HCLK	AHB Hready, for dedicated Initiator-Target channel
TXHBURST	3	Output	HCLK	AHB Burst transfers **Note:** This port is tied to 3'b001, that is, core supports only incrementing burst of undefined length burst type.
TXHLOCK	1	Output	HCLK	Locked access to the busn **Note:** This port is tied to 1'b0, that is, core does not support lock transfer.
TXHSIZE	3	Output	HCLK	Size of the transfer **Note:** This port is tied to 2'b10, that is, core supports only 32-bit size operation.
Rx-DMA AHB-Initiator Interface Signals				
RXHGRANT	1	Input	HCLK	AHB bus grant
RXHRESP	2	Input	HCLK	AHB response
RXHRDATA	32	Input	HCLK	AHB read data
RXHBUSREQ	1	Output	HCLK	AHB Initiator bus request
RXHTRANS	2	Output	HCLK	AHB Initiator transaction type
RXHADDR	32	Output	HCLK	AHB Initiator transaction address
RXHWRITE	1	Output	HCLK	AHB Initiator transfer direction
RXHWDATA	32	Output	HCLK	AHB Initiator write data bus
RXHREADY	1	Input	HCLK	AHB Hready, for dedicated Initiator-Target channel
RXHBURST	3	Output	HCLK	AHB Burst transfers **Note:** This port is tied to 3'b001, that is, core supports only incrementing burst of undefined length burst type.
RXHLOCK	1	Output	HCLK	Locked access to the bus **Note:** This port is tied to 1'b0, that is, core does not support lock transfer.
RXHSIZE	3	Output	HCLK	Size of the transfer **Note:** This port is tied to 2'b10, that is, core supports only 32-bit size operation.
Misc Signals				
TSM_INTR	4	Output	HCLK	**Common Interrupt port** Providing these individual interrupts at top allows the user to connect required interrupts to the host-processor based on application requirements. This port is visible to the user only when **"Enable Separate Tx and Rx Interrupt ports"** parameter is enabled in the GUI configurator. **Bit [3]:** ECC error Interrupt – This includes Tx and embedded RAM's SEC & DED errors. **Bit [2]:** Wake on LAN detected interrupt **Bit [1]:** Statistics counter carry interrupt Statistics counter carry interrupt is generated whenever there is a carry produced in any of the implemented counters This is deasserted whenever the user clears the generated carry bits in the CAR1 and CAR2 registers. **Bit [0]:** DMA Tx and Rx Interrupt

Table 4-4. Input and Output Ports (continued)

Port Name	Width	Direction	Clock Domain	Description
TSM_TX_INTR	4	Output	HCLK	**Transmitter Interrupt port** This port is enabled only when "**Enable Separate Tx and Rx Interrupt ports"** parameter is enabled in the GUI configurator This port has all the interrupt signals related to Tx. **Bit [3]:** Error Interrupt – This includes all DMA related errors, SEC and DED errors of the embedded RAM used in transmitter path. Interrupt for TX SEC/DED are asserted when the respective TX embedded RAM flags gets asserted and are deasserted when there is a read request for the or TDBEDC registers. **Bit [2]:** Reserved (Default: 1'b0) **Bit [1]:** Tx Statistics counters carry interrupt Tx Statistics counter carry interrupt is generated whenever there is a carry produced in any of the implemented counters for Tx.This is deasserted whenever the user clears the generated carry bits in the CAR2 register. **Bit [0]:** Tx DMA Interrupt
TSM_RX_INTR	4	Output	HCLK	**Receiver Interrupt port** This port is visible to you only when the **Enable Separate Tx and Rx Interrupt ports** parameter is enabled in the GUI configurator. This port has all the interrupt signals related to Rx. **Bit [3]:** Error Interrupt This includes all DMA related errors, SEC and DED errors of embedded RAMs used in Receiver path. Interrupt for RX SEC/DED is asserted when the respective RX embedded RAMflags gets asserted and is deasserted when there is a read request for the RSBECC / RDBEDC registers. **Bit [2]:** Wake on LAN detected interrupt **Bit [1]:** RX Statistics counters carry interrupt.RX Statistics counter carry interrupt is generated whenever there is a carry produced in any of the implemented counters related to Rx. It is deasserted whenever the user clears the generated carry bits in the CAR1 register. **Bit [0]:** Rx DMA Interrupt
TSM_CONTROL	32	Output	HCLK	32-bit GPIO output signals mapped to system miscellaneous control register (0x1D4)
RCG_ERROR	1	Output	TBI_RX_CLK	Indicates the receive code group error

5. Register Map and Descriptions (Ask a Question)

This section provides the register map descriptions for MAC and MDIO.

5.1. Register Map Section Overview Table (Ask a Question)

The external AHB initiator uses a 32-bit AHB target interface for accessing control and status registers.

Table 5-1. CoreTSE_AHB Register Map

Address Offset	Function
0x000 – 0x044	Access to MAC core registers
0x048 – 0x07C	Access to FIFO core registers
0x080 – 0x17F	Access to Statistics Counters core registers
0x180 – 0x1BF	Access to AHB-DMA registers
0x1C0 – 0x1FF	Access to System Registers (SAL and miscellaneous controls) 0x1C0 – 0x1D4 are valid addresses

5.2. MAC Register Summary (Ask a Question)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	MACConfiguration1	7:0			RECEIVE FLOW CONTROL ENABLE	TRANSMIT FLOW CONTROL ENABLE	SYNCHRONIZED RECEIVE ENABLE	RECEIVE ENABLE	SYNCHRONIZED TRANSMIT ENABLE	TRANSMIT ENABLE
		15:8								LOOP BACK
		23:16					RESET RX MAC CONTROL	RESET TX MAC CONTROL	RESET RX FUNCTION	RESET TX FUNCTION
		31:24	SOFT RESET	SIMULATION RESET						
0x04	MACConfiguration2	7:0		RX CRC DISABLE	HUGE FRAME ENABLE	LENGTH FIELD CHECKING		PAD/CRC ENABLE	CRC ENABLE	FULL-DUPLEX
		15:8	PREAMBLE LENGTH[3:0]						INTERFACE MODE[1:0]	
		23:16								
		31:24								
0x08	IPG	7:0		BACK_TO_BACK INTER_PACKET_GAP[6:0]						
		15:8	MINIMUM IFG ENFORCEMENT[7:0]							
		23:16		NON_BACK_TO_BACK INTER_PACKET_GAP PART2 (IPGR2)[6:0]						
		31:24		NON_BACK_TO_BACK INTER_PACKET_GAP PART1 (IPGR1)[6:0]						
0x0C	Half-Duplex	7:0	COLLISION WINDOW[7:0]							
		15:8	RETRANSMISSION MAXIMUM[3:0]						COLLISION WINDOW[9:8]	
		23:16	ALTERNATE BINARY EXPONENTIAL BACKOFF TRUNCATION[3:0]				ALTERNATE BINARY EXPONENTIAL BACKOFF ENABLE	BACKPRESSURE NO BACKOFF	NO BACKOFF	EXCESSIVE DEFER
		31:24								
0x10	Maximum_Frame_Length	7:0	MAXIMUM FRAME LENGTH[7:0]							
		15:8	MAXIMUM FRAME LENGTH[15:8]							
		23:16								
		31:24								
0x14	Control_Frame_extended_parameter	7:0	CFEP[7:0]							
		15:8	CFEP[15:8]							
		23:16								
		31:24								
0x18	Control_Frame_parameter	7:0	CFPT[7:0]							
		15:8	CFPT[15:8]							
		23:16								
		31:24								
0x1C	Test_Register	7:0					MAXIMUM BACKOFF	REGISTERED TRANSMIT FLOW ENABL	TEST PAUSE	SHORTCUT SLOT TIME
		15:8								
		23:16								
		31:24								
0x20	MIIMgmt_Configuration	7:0			SCAN AUTO INCREMENT	PREAMBLE SUPPRESSION		MGMT CLOCK SELECT[2:0]		
		15:8								
		23:16								
		31:24	RESET MDIO MGMT							
0x24	MIIMgmt_Command	7:0							SCAN CYCLE	READ CYCLE
		15:8								
		23:16								
		31:24								

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x28	MIIMgmt_Address	7:0				REGISTER ADDRESS[4:0]				
		15:8				PHY ADDRESS[4:0]				
		23:16								
		31:24								
0x2C	MIIMgmt_Control	7:0	MDIO MGMT CONTROL (PHY CONTROL)[7:0]							
		15:8	MDIO MGMT CONTROL (PHY CONTROL)[15:8]							
		23:16								
		31:24								
0x30	MIIMgmt_Status	7:0	MDIO MGMT STATUS (PHY STATUS)[7:0]							
		15:8	MDIO MGMT STATUS (PHY STATUS)[15:8]							
		23:16								
		31:24								
0x34	MIIMgmt_Indicators	7:0						NOT VALID	SCANNING	BUSY
		15:8								
		23:16								
		31:24								
0x38	Interface_Control	7:0	WOL: UNICAST MATCH ENABLE	WOL: MAGIC PACKET DETECTION ENABLE	WOL: WAKE ON LANE DETECTED CLEAR STATUS CLEAR	STATS COUNTERS : AUTO CLEAR COUNTERS ON READ	STATS COUNTERS : CLEAR ALL COUNTERS	STATS COUNTERS : MODULE ENABLE		
		15:8								
		23:16								
		31:24								
0x3C	Interface_Status	7:0					LINK FAIL			
		15:8						WAKE ON LANE DETECTED	EXCESS DEFER	
		23:16								
		31:24								
0x40	Station_Address_Lower_Register	7:0	FOURTH OCTET OF THE DA IN THE FRAME[7:0]							
		15:8	THIRD OCTET OF THE DA IN THE FRAME[7:0]							
		23:16	SECOND OCTET OF THE DA IN THE FRAME[7:0]							
		31:24	FIRST OCTET OF THE DA IN THE FRAME[7:0]							
0x44	Station_Address_Higher_Register	7:0								
		15:8								
		23:16	SIXTH OCTET OF THE DA IN THE FRAME[7:0]							
		31:24	FIFTH OCTET OF THE DA IN THE FRAME[7:0]							
0x48	A-MCXFIF0	7:0				HOST FABRIC TRANSMIT MODULE RESET (HSTRSTFT)	HOST MAC TRANSMIT MODULE RESET (HSTRSTST)	HOST FABRIC RECEIVE MODULE RESET (HSTRSTFR)	HOST MAC RECEIVE MODULE RESET (HSTRSTSR)	HOST TRANSMIT WATERMARK MODULE RESET (HSTRSTWT)
		15:8				FABRIC TRANSMIT MODULE ENABLE REQUEST (FTFENREQ)	SYSTEM TRANSMIT MODULE ENABLE REQUEST (STFENREQ)	FABRIC RECEIVE MODULE ENABLE REQUEST (FRFENREQ)	SYSTEM RECEIVE MODULE ENABLE REQUEST(SRFENREQ)	WATER MARK MODULE ENABLE REQUEST(WTMENREQ)
		23:16				FABRIC TRANSMIT MODULE ENABLE STATUS (FTFENRPLY)	SYSTEM TRANSMIT MODULE ENABLE STATUS (STFENRPLY)	FABRIC RECEIVE MODULE ENABLE STATUS (FRFENRPLY)	SYSTEM RECEIVE MODULE ENABLE STATUS (SRFENRPLY)	WATER MARK MODULE ENABLE STATUS (WTMENRPLY)
		31:24								

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x4C	A-MCXFIF1	7:0	NUMBER OF PAUSE QUANTAS BEFORE XOFF RETRANSMISSION (CFGXOFFRTX) [7:0]							
		15:8	NUMBER OF PAUSE QUANTAS BEFORE XOFF RETRANSMISSION (CFGXOFFRTX) [15:8]							
		23:16	SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH) [7:0]							
		31:24					SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH) [11:8]			
0x50	A-MCXFIF2	7:0	MIN WORDS IN RECEIVE FIFO BEFORE (CFGLWM) [7:0]							
		15:8				MIN WORDS IN RECEIVE FIFO BEFORE (CFGLWM) [12:8]				
		23:16	MAX WORDS IN RECEIVE FIFO (CFGHWM) [7:0]							
		31:24				MAX WORDS IN RECEIVE FIFO (CFGHWM) [12:8]				
0x54	A-MCXFIF3	7:0	FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH) [7:0]							
		15:8				FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH) [12:8]				
		23:16	MAX NUMBER OF WORDS IN TRANSMIT FIFO (CFGHWMFT) [7:0]							
		31:24					MAX NUMBER OF WORDS IN TRANSMIT FIFO (CFGHWMFT) [11:8]			
0x58	A-MCXFIF4	7:0	HOST FILTER FRAMES (HSTFLTRFRM) [7:0]							
		15:8	HOST FILTER FRAMES (HSTFLTRFRM) [15:8]							
		23:16							HOST FILTER FRAMES (HSTFLTRFRM) [17:16]	
		31:24								
0x5C	A-MCXFIF5	7:0	HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [7:0]							
		15:8	HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [15:8]							
		23:16		HALF DUPLEX INDICATOR (CFGHDPLX)	SYSTEM RECEIVE FIFO FULL (SRFULL)	HOST CLEAR SYSTEM RECEIVE FIFO FULL (HSTSRFULLCLR)	ONE BYTE TRANSFER PER SYSTEM CLOCK (CFGBYTMODE)	HOST DROP FRAMES LESS THAN 64 BYTES (HSTDRPLT64)	HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [17:16]	
		31:24								
0x60	A-MCXFIFRAMAccess0	7:0	HOST TRANSMIT RAM WRITE ADDRESS (HSTTRAMWADX [12:0])							
		15:8				HOST TRANSMIT RAM WRITE ADDRESS (HSTTRAMWADX [12:0])				
		23:16	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[39:32])							
		31:24	HOST TRANSMIT RAM WRITE REQUEST (HSTTRAMWREQ)	HOST TRANSMIT RAM WRITE ACKNOWLEDGE (HSTTRAMWACK)						
0x64	A-MCXFIFRAMAccess1	7:0	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
		15:8	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
		23:16	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
		31:24	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
0x68	A-MCXFIFRAMAccess2	7:0	HOST TRANSMIT RAM READ ADDRESS (HSTTRAMRADX[12:0])							
		15:8				HOST TRANSMIT RAM READ ADDRESS (HSTTRAMRADX[12:0])				
		23:16	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[39:32])							
		31:24	HOST TRANSMIT RAM READ REQUEST (HSTTRAMRREQ)	HOST TRANSMIT RAM READ ACKNOWLEDGE (HSTTRAMRACK)						
0x6C	A-MCXFIFRAMAccess3	7:0	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
		15:8	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
		23:16	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
		31:24	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x70	A-MCXFIFRAMAccess4	7:0	HOST RECEIVE RAM WRITE ADDRESS (HSTTRAMWADX[(RABITS+2):0])							
		15:8		HOST RECEIVE RAM WRITE ADDRESS (HSTTRAMWADX[(RABITS+2):0])						
		23:16	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT[39:32])							
		31:24	HOST RECEIVE RAM WRITE REQUEST (HSTRRAMWREQ)	HOST RECEIVE RAM WRITE ACKNOWLEDGE (HSTRRAMWACK)						
0x74	A-MCXFIFRAMAccess5	7:0	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
		15:8	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
		23:16	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
		31:24	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
0x78	A-MCXFIFRAMAccess6	7:0	HOST RECEIVE RAM READ ADDRESS (HSTRRAMRADX[31:0])							
		15:8			HOST RECEIVE RAM READ ADDRESS (HSTRRAMRADX[31:0])					
		23:16	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[39:32])							
		31:24	HOST RECEIVE RAM READ REQUEST (HSTRRAMRREQ)	HOST RECEIVE RAM READ ACKNOWLEDGE (HSTRRAMRACK)						
0x7C	A-MCXFIFRAMAccess7	7:0	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
		15:8	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
		23:16	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
		31:24	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
0x80	TR64	7:0	TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [17:16]	
		31:24								
0x84	TR127	7:0	TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [17:16]	
		31:24								
0x88	TR255	7:0	TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [17:16]	
		31:24								
0x8C	TR511	7:0	TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [17:16]	
		31:24								
0x90	TR1K	7:0	TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [17:16]	
		31:24								

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x94	TRMAX	7:0	TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [17:16]	
		31:24								
0x98	TRMGV	7:0	TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [7:0]							
		15:8	TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [15:8]							
		23:16							TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [17:16]	
		31:24								
0x9C	RBYT	7:0	RECEIVE BYTE COUNTER [7:0]							
		15:8	RECEIVE BYTE COUNTER [15:8]							
		23:16	RECEIVE BYTE COUNTER [23:16]							
		31:24								
0xA0	RPKT	7:0	RECEIVE PACKET COUNTER [7:0]							
		15:8	RECEIVE PACKET COUNTER [15:8]							
		23:16							RECEIVE PACKET COUNTER [17:16]	
		31:24								
0xA4	RFCS	7:0	RECEIVE FCS ERROR COUNTER [7:0]							
		15:8					RECEIVE FCS ERROR COUNTER [11:8]			
		23:16								
		31:24								
0xA8	RMCA	7:0	RECEIVE MULTICAST PACKET COUNTER [7:0]							
		15:8	RECEIVE MULTICAST PACKET COUNTER [15:8]							
		23:16							RECEIVE MULTICAST PACKET COUNTER [17:16]	
		31:24								
0xAC	RBCA	7:0	RECEIVE BROADCAST PACKET COUNTER [7:0]							
		15:8	RECEIVE BROADCAST PACKET COUNTER [15:8]							
		23:16							RECEIVE BROADCAST PACKET COUNTER [17:16]	
		31:24								
0xB0	RXCF	7:0	RECEIVE CONTROL FRAME PACKET COUNTER [7:0]							
		15:8					RECEIVE CONTROL FRAME PACKET COUNTER [11:8]			
		23:16								
		31:24								
0xB4	RXPF	7:0	RECEIVE PAUSE CONTROL FRAME COUNTER [7:0]							
		15:8					RECEIVE PAUSE CONTROL FRAME COUNTER [11:8]			
		23:16								
		31:24								
0xB8	RXUO	7:0	RECEIVE UNKNOWN OPCODE PACKET COUNTER [7:0]							
		15:8					RECEIVE UNKNOWN OPCODE PACKET COUNTER [11:8]			
		23:16								
		31:24								
0xBC	RALN	7:0	RECEIVE ALIGNMENT ERROR COUNTER [7:0]							
		15:8					RECEIVE ALIGNMENT ERROR COUNTER [11:8]			
		23:16								
		31:24								
0xC0	RFLR	7:0	RECEIVE FRAME LENGTH ERROR COUNTER [7:0]							
		15:8					RECEIVE FRAME LENGTH ERROR COUNTER [11:8]			
		23:16								
		31:24								
0xC4	RCDE	7:0	RECEIVE CODE ERROR COUNTER [7:0]							
		15:8					RECEIVE CODE ERROR COUNTER [11:8]			
		23:16								
		31:24								

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0xC8	RCSE	7:0	RECEIVE CARRIER SENSE ERROR COUNTER [7:0]							
		15:8					RECEIVE CARRIER SENSE ERROR COUNTER [11:8]			
		23:16								
		31:24								
0xCC	RUND	7:0	RECEIVE UNDERSIZE PACKET COUNTER [7:0]							
		15:8					RECEIVE UNDERSIZE PACKET COUNTER [11:8]			
		23:16								
		31:24								
0xD0	ROVR	7:0	RECEIVE OVERSIZE PACKET COUNTER [7:0]							
		15:8					RECEIVE OVERSIZE PACKET COUNTER [11:8]			
		23:16								
		31:24								
0xD4	RFRG	7:0	RECEIVE FRAGMENTS COUNTER [7:0]							
		15:8					RECEIVE FRAGMENTS COUNTER [11:8]			
		23:16								
		31:24								
0xD8	RJBR	7:0	RECEIVE JABBER COUNTER [7:0]							
		15:8					RECEIVE JABBER COUNTER [11:8]			
		23:16								
		31:24								
0xDC	RDRP	7:0	RECEIVE DROP [7:0]							
		15:8					RECEIVE DROP [11:8]			
		23:16								
		31:24								
0xE0	TBYT	7:0	TRANSMIT BYTE COUNTER [7:0]							
		15:8	TRANSMIT BYTE COUNTER [15:8]							
		23:16	TRANSMIT BYTE COUNTER [23:16]							
		31:24								
0xE4	TPKT	7:0	TRANSMIT PACKET COUNTER [7:0]							
		15:8	TRANSMIT PACKET COUNTER [15:8]							
		23:16							TRANSMIT PACKET COUNTER [17:16]	
		31:24								
0xE8	TMCA	7:0	TRANSMIT MULTICAST PACKET COUNTER [7:0]							
		15:8	TRANSMIT MULTICAST PACKET COUNTER [15:8]							
		23:16							TRANSMIT MULTICAST PACKET COUNTER [17:16]	
		31:24								
0xEC	TBCA	7:0	TRANSMIT BROADCAST PACKET COUNTER [7:0]							
		15:8	TRANSMIT BROADCAST PACKET COUNTER [15:8]							
		23:16							TRANSMIT BROADCAST PACKET COUNTER [17:16]	
		31:24								
0xF0	TXPF	7:0	TRANSMIT PAUSE CONTROL FRAME COUNTER [7:0]							
		15:8					TRANSMIT PAUSE CONTROL FRAME COUNTER [11:8]			
		23:16								
		31:24								
0xF4	TDFR	7:0	TRANSMIT DEFERRAL PACKET COUNTER [7:0]							
		15:8					TRANSMIT DEFERRAL PACKET COUNTER [11:8]			
		23:16								
		31:24								
0xF8	TEDF	7:0	TRANSMIT EXCESSIVE DEFERRAL PACKET COUNTER [7:0]							
		15:8					TRANSMIT EXCESSIVE DEFERRAL PACKET COUNTER [11:8]			
		23:16								
		31:24								
0xFC	TSCL	7:0	TRANSMIT SINGLE COLLISION PACKET COUNTER [7:0]							
		15:8					TRANSMIT SINGLE COLLISION PACKET COUNTER [11:8]			
		23:16								
		31:24								

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x0100	TMCL	7:0	TRANSMIT MULTIPLE COLLISION PACKET COUNTER [7:0]							
		15:8					TRANSMIT MULTIPLE COLLISION PACKET COUNTER [11:8]			
		23:16								
		31:24								
0x0104	TLCL	7:0	TRANSMIT LATE COLLISION PACKET COUNTER [7:0]							
		15:8					TRANSMIT LATE COLLISION PACKET COUNTER [11:8]			
		23:16								
		31:24								
0x0108	TXCL	7:0	TRANSMIT EXCESSIVE COLLISION PACKET COUNTER [7:0]							
		15:8					TRANSMIT EXCESSIVE COLLISION PACKET COUNTER [11:8]			
		23:16								
		31:24								
0x010C	TNCL	7:0	TRANSMIT TOTAL COLLISION COUNTER [7:0]							
		15:8					TRANSMIT TOTAL COLLISION COUNTER [11:8]			
		23:16								
		31:24								
0x0110 ... 0x0113	Reserved									
0x0114	TDRP	7:0	TRANSMIT DROP FRAME COUNTER [7:0]							
		15:8					TRANSMIT DROP FRAME COUNTER [11:8]			
		23:16								
		31:24								
0x0118	TJBR	7:0	TRANSMIT JABBER FRAME COUNTER [7:0]							
		15:8					TRANSMIT JABBER FRAME COUNTER [11:8]			
		23:16								
		31:24								
0x011C	TFCS	7:0	TRANSMIT FCS ERROR COUNTER [7:0]							
		15:8					TRANSMIT FCS ERROR COUNTER [11:8]			
		23:16								
		31:24								
0x0120	TXCF	7:0	TRANSMIT CONTROL FRAME COUNTER [7:0]							
		15:8					TRANSMIT CONTROL FRAME COUNTER [11:8]			
		23:16								
		31:24								
0x0124	TOVR	7:0	TRANSMIT OVERSIZE FRAME COUNTER [7:0]							
		15:8					TRANSMIT OVERSIZE FRAME COUNTER [11:8]			
		23:16								
		31:24								
0x0128	TUND	7:0	TRANSMIT UNDER SIZE FRAME COUNTER [7:0]							
		15:8					TRANSMIT UNDER SIZE FRAME COUNTER [11:8]			
		23:16								
		31:24								
0x012C	TFRG	7:0	TRANSMIT FRAGMENTS FRAME COUNTER [7:0]							
		15:8					TRANSMIT FRAGMENTS FRAME COUNTER [11:8]			
		23:16								
		31:24								

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x0130	CAR1	7:0	CARRY REGISTER 1 RFLR COUNTER CARRY BIT	CARRY REGISTER 1 RCDE COUNTER CARRY BIT	CARRY REGISTER 1 RCSE COUNTER CARRY BIT	CARRY REGISTER 1 RUND COUNTER CARRY BIT	CARRY REGISTER 1 ROVR COUNTER CARRY BIT	CARRY REGISTER 1 RFRG COUNTER CARRY BIT	CARRY REGISTER 1 RJBR COUNTER CARRY BIT	CARRY REGISTER 1 RDRP COUNTER CARRY BIT
		15:8	CARRY REGISTER 1 RPKT COUNTER CARRY BIT	CARRY REGISTER 1 RFCS COUNTER CARRY BIT	CARRY REGISTER 1 RMCA COUNTER CARRY BIT	CARRY REGISTER 1 RBCA COUNTER CARRY BIT	CARRY REGISTER 1 RXCF COUNTER CARRY BIT	CARRY REGISTER 1 RXPF COUNTER CARRY BIT	CARRY REGISTER 1 RXUO COUNTER CARRY BIT	CARRY REGISTER 1 RALN COUNTER CARRY BIT
		23:16						CARRY REGISTER 1 RSBECC COUNTER CARRY BIT	CARRY REGISTER 1 RDBEDC COUNTER CARRY BIT	CARRY REGISTER 1 RBYT COUNTER CARRY BIT
		31:24	CARRY REGISTER 1 TR64 COUNTER CARRY BIT	CARRY REGISTER 1 TR127 COUNTER CARRY BIT	CARRY REGISTER 1 TR255 COUNTER CARRY BIT	CARRY REGISTER 1 TR511 COUNTER CARRY BIT	CARRY REGISTER 1 TR1K COUNTER CARRY BIT	CARRY REGISTER 1 TRMAX COUNTER CARRY BIT	CARRY REGISTER 1 TRMGV COUNTER CARRY BIT	
0x0134	CAR2	7:0	CARRY REGISTER 2 TEDF COUNTER CARRY BIT	CARRY REGISTER 2 TSCL COUNTER CARRY BIT	CARRY REGISTER 2 TMCL COUNTER CARRY BIT	CARRY REGISTER 2 TLCL COUNTER CARRY BIT	CARRY REGISTER 2 TXCL COUNTER CARRY BIT	CARRY REGISTER 2 TNCL COUNTER CARRY BIT	CARRY REGISTER 2 TPFH COUNTER CARRY BIT	CARRY REGISTER 2 TDRP COUNTER CARRY BIT
		15:8	CARRY REGISTER 2 TUND COUNTER CARRY BIT	CARRY REGISTER 2 TFRG COUNTER CARRY BIT	CARRY REGISTER 2 TBYT COUNTER CARRY BIT	CARRY REGISTER 2 TPKT COUNTER CARRY BIT	CARRY REGISTER 2 TMCA COUNTER CARRY BIT	CARRY REGISTER 2 TBCA COUNTER CARRY BIT	CARRY REGISTER 2 TXPF COUNTER CARRY BIT	CARRY REGISTER 2 TDFR COUNTER CARRY BIT
		23:16			CARRY REGISTER 2 TSBECC COUNTER CARRY BIT	CARRY REGISTER 2 TDBEDC COUNTER CARRY BIT	CARRY REGISTER 2 TJBR COUNTER CARRY BIT	CARRY REGISTER 2 TXFC COUNTER CARRY BIT	CARRY REGISTER 2 TXCF COUNTER CARRY BIT	CARRY REGISTER 2 TOVR COUNTER CARRY BIT
		31:24								
0x0138	CAM1	7:0	MASK REGISTER 1 RFLR COUNTER CARRY BIT	MASK REGISTER 1 RCDE COUNTER CARRY BIT	MASK REGISTER 1 RCSE COUNTER CARRY BIT	MASK REGISTER 1 RUND COUNTER CARRY BIT	MASK REGISTER 1 ROVR COUNTER CARRY BIT	MASK REGISTER 1 RFRG COUNTER CARRY BIT	MASK REGISTER 1 RJBR COUNTER CARRY BIT	MASK REGISTER 1 RDRP COUNTER CARRY BIT
		15:8	MASK REGISTER 1 RPKT COUNTER CARRY BIT	MASK REGISTER 1 RFCS COUNTER CARRY BIT	MASK REGISTER 1 RMCA COUNTER CARRY BIT	MASK REGISTER 1 RBCA COUNTER CARRY BIT	MASK REGISTER 1 RXCF COUNTER CARRY BIT	MASK REGISTER 1 RXPF COUNTER CARRY BIT	MASK REGISTER 1 RXUO COUNTER CARRY BIT	MASK REGISTER 1 RALN COUNTER CARRY BIT
		23:16						MASK REGISTER 1 RSBECC COUNTER CARRY BIT	MASK REGISTER 1 RDBEDC COUNTER CARRY BIT	MASK REGISTER 1 RBYT COUNTER CARRY BIT
		31:24	MASK REGISTER 1 TR64 COUNTER CARRY BIT	MASK REGISTER 1 TR127 COUNTER CARRY BIT	MASK REGISTER 1 TR255 COUNTER CARRY BIT	MASK REGISTER 1 TR511 COUNTER CARRY BIT	MASK REGISTER 1 TR1K COUNTER CARRY BIT	MASK REGISTER 1 TRMAX COUNTER CARRY BIT	MASK REGISTER 1 TRMGV COUNTER CARRY BIT	

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x013C	CAM2	7:0	MASK REGISTER 2 TEDF COUNTER CARRY BIT	MASK REGISTER 2 TSCL COUNTER CARRY BIT	MASK REGISTER 2 TMCL COUNTER CARRY BIT	MASK REGISTER 2 TLCL COUNTER CARRY BIT	MASK REGISTER 2 TXCL COUNTER CARRY BIT	MASK REGISTER 2 TNCL COUNTER CARRY BIT	MASK REGISTER 2 TPFH COUNTER CARRY BIT	MASK REGISTER 2 TDRP COUNTER CARRY BIT
		15:8	MASK REGISTER 2 TUND COUNTER CARRY BIT	MASK REGISTER 2 TFRG COUNTER CARRY BIT	MASK REGISTER 2 TBYT COUNTER CARRY BIT	MASK REGISTER 2 TPKT COUNTER CARRY BIT	MASK REGISTER 2 TMCA COUNTER CARRY BIT	MASK REGISTER 2 TBCA COUNTER CARRY BIT	MASK REGISTER 2 TXPF COUNTER CARRY BIT	MASK REGISTER 2 TDFR COUNTER CARRY BIT
		23:16			MASK REGISTER 2 TSBECC COUNTER CARRY BIT	MASK REGISTER 2 TDBEDC COUNTER CARRY BIT	MASK REGISTER 2 TJBR COUNTER CARRY BIT	MASK REGISTER 2 TXFC COUNTER CARRY BIT	MASK REGISTER 2 TXCF COUNTER CARRY BIT	MASK REGISTER 2 TOVR COUNTER CARRY BIT
		31:24								
0x0140	TSBECC	7:0	TRANSMIT ECC SEC COUNTER [7:0]							
		15:8	TRANSMIT ECC SEC COUNTER [15:8]							
		23:16								
		31:24								
0x0144	TDBEDC	7:0	TRANSMIT ECC DED COUNTER [7:0]							
		15:8	TRANSMIT ECC DED COUNTER [15:8]							
		23:16								
		31:24								
0x0148	RSBECC	7:0	RECEIVER ECC SEC COUNTER [7:0]							
		15:8	RECEIVER ECC SEC COUNTER [15:8]							
		23:16								
		31:24								
0x014C	RDBEDC	7:0	RECEIVER ECC DED COUNTER [7:0]							
		15:8	RECEIVER ECC DED COUNTER [15:8]							
		23:16								
		31:24								
0x0150 ... 0x017F	Reserved									
0x0180	DMATxCtrl	7:0								TX ENABLE
		15:8	TX INTERRUPT COALESCING THRESHOLD COUNT[7:0]							
		23:16								
		31:24								
0x0184	DMATxDescriptor	7:0	TOP 30 BITS OF DESCRIPTOR ADDRESS[5:0]						IGNORED[1:0]	
		15:8	TOP 30 BITS OF DESCRIPTOR ADDRESS[13:6]							
		23:16	TOP 30 BITS OF DESCRIPTOR ADDRESS[21:14]							
		31:24	TOP 30 BITS OF DESCRIPTOR ADDRESS[29:22]							
0x0188	DMATxStatus	7:0					TXBUSERROR		TXUNDERRUN	TXPKTSENT
		15:8								
		23:16	TXPKTCOUNT[7:0]							
		31:24							TX RAM SINGLE BIT ERROR DETECTED AND CORRECTED	TX RAM DOUBLE BIT ERROR DETECTED
0x018C	DMARxCtrl	7:0								RX ENABLE
		15:8	RX INTERRUPT COALESCING THRESHOLD COUNT[7:0]							
		23:16								
		31:24								
0x0190	DMARxDescriptor	7:0	TOP 30 BITS OF DESCRIPTOR ADDRESS[5:0]						IGNORED[1:0]	
		15:8	TOP 30 BITS OF DESCRIPTOR ADDRESS[13:6]							
		23:16	TOP 30 BITS OF DESCRIPTOR ADDRESS[21:14]							
		31:24	TOP 30 BITS OF DESCRIPTOR ADDRESS[29:22]							

MAC Register Summary (continued)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x0194	DMARxStatus	7:0					RXBUSERROR	RXOVERFLOW		RXPKTRECEIVED
		15:8								
		23:16	RXPKTCOUNT[7:0]							
		31:24							RX RAM SINGLE BIT ERROR DETECTED AND CORRECTED	RX RAM DOUBLE BIT ERROR DETECTED
0x0198	DMAIntrMask Interrupt Mask	7:0	BUS ERROR MASK	RX OVERFLOW MASK		RXPKTRECEIVED MASK	TX BUS ERROR MASK		TX UNDERRUN MASK	TXPKTSENT MASK
		15:8			TX_ECC_SEC INTERRUPT MASK	TX_ECC_DED INTERRUPT MASK	RX_ECC_SEC INTERRUPT MASK	RX_ECC_DED INTERRUPT MASK	RXPKT INTERRUPT COALESCING MASK	TXPKT INTERRUPT COALESCING MASK
		23:16								
		31:24								
0x019C	DMAInterrupt Interrupt status	7:0	RX BUS ERROR	RX OVERFLOW		RXPKTRECEIVED	BUS ERROR		TX UNDERRUN	TXPKTSENT
		15:8			TX_ECC_SEC INTERRUPT STATUS	TX_ECC_DED INTERRUPT STATUS	RX_ECC_SEC INTERRUPT STATUS	RX_ECC_DED INTERRUPT STATUS	RXPKT INTERRUPT COALESCING	TXPKT INTERRUPT COALESCING
		23:16								
		31:24								
0x01A0 ... 0x01BF	Reserved									
0x01C0	Framefiltercontrols	7:0			FRAME FILTER CONTROLS[5:0]					
		15:8								
		23:16								
		31:24								
0x01C4	Hash_Table_Register0	7:0	HASH TABLE REGISTER0 [7:0]							
		15:8	HASH TABLE REGISTER0 [15:8]							
		23:16	HASH TABLE REGISTER0 [23:16]							
		31:24	HASH TABLE REGISTER0 [31:24]							
0x01C8	Hash_Table_Register1	7:0	HASH TABLE REGISTER1 [7:0]							
		15:8	HASH TABLE REGISTER1 [15:8]							
		23:16	HASH TABLE REGISTER1 [23:16]							
		31:24	HASH TABLE REGISTER1 [31:24]							
0x01CC	Hash_Table_Register2	7:0	HASH TABLE REGISTER2 [7:0]							
		15:8	HASH TABLE REGISTER2 [15:8]							
		23:16	HASH TABLE REGISTER2 [23:16]							
		31:24	HASH TABLE REGISTER2 [31:24]							
0x01D0	Hash_Table_Register3	7:0	HASH TABLE REGISTER3 [7:0]							
		15:8	HASH TABLE REGISTER3 [15:8]							
		23:16	HASH TABLE REGISTER3 [23:16]							
		31:24	HASH TABLE REGISTER3 [31:24]							
0x01D4	Misc_Control_register	7:0								
		15:8								
		23:16								
		31:24								

5.2.1. TXPF (Ask a Question)

Name: TXPF
Offset: 0x0F0
Reset: 0x0
Property: Read-only

TXPF- Transmit PAUSE Control Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT PAUSE CONTROL FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT PAUSE CONTROL FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT PAUSE CONTROL FRAME COUNTER [11:0] Incremented each time a valid PAUSE MAC Control frame is transmitted.

5.2.2. TXCL (Ask a Question)

Name: TXCL
Offset: 0x108
Reset: 0x0
Property: Read-only

TXCL- Transmit Excessive Collision Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT EXCESSIVE COLLISION PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT EXCESSIVE COLLISION PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT EXCESSIVE COLLISION PACKET COUNTER [11:0] Incremented for each frame that experienced 16 collisions during transmission and was aborted.

5.2.3. TXCF (Ask a Question)

Name: TXCF
Offset: 0x120
Reset: 0x0
Property: Read-only

TXCF- Transmit Control Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT CONTROL FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT CONTROL FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT CONTROL FRAME COUNTER [11:0] Incremented for every valid size frame with a Type Field signifying a Control frame.

5.2.4. TUND (Ask a Question)

Name: TUND
Offset: 0x128
Reset: 0x0
Property: Read-only

TUND- Transmit Under size Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT UNDER SIZE FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT UNDER SIZE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT UNDER SIZE FRAME COUNTER [11:0] Incremented for every frame less than 64 bytes, with a correct FCS value.

5.2.5. TSCL (Ask a Question)

Name: TSCL
Offset: 0x0FC
Reset: 0x0
Property: Read-only

TSCL- Transmit Single Collision Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT SINGLE COLLISION PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT SINGLE COLLISION PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT SINGLE COLLISION PACKET COUNTER [11:0] Incremented for each frame transmitted which experienced exactly one collision during transmission.

5.2.6. TSBECC (Ask a Question)

Name: TSBECC
Offset: 0x140
Reset: 0x0
Property: Read-only

TSBECC Transmit ECC SEC Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	TRANSMIT ECC SEC COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT ECC SEC COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – TRANSMIT ECC SEC COUNTER [15:0] Incremented this counter whenever single bit error correct (SB_CORRECT) flag of transmit buffer memory is asserted.

5.2.7. TRMGV (Ask a Question)

Name: TRMGV
Offset: 0x098
Reset: 0x0
Property: Read-only

TRMGV-Transmit & Receive 1519 to 1522 Byte VLAN Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 1519 TO 1522 BYTE VLAN FRAME COUNTER [17:0] Incremented for each good VLAN frame transmitted and received which is 1519 to 1522 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.8. TRMAX (Ask a Question)

Name: TRMAX
Offset: 0x094
Reset: 0x0
Property: Read-only

TRMAX - Transmit & Receive 1024 to 1518 Byte Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 1024 TO 1518 BYTE FRAME COUNTER [17:0] Incremented for each good or bad frame transmitted and received which is 1024 to 1518 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.9. TR64 (Ask a Question)

Name: TR64
Offset: 0x080
Reset: 0x0
Property: Read-only

TR64 - Transmit & Receive 64 Byte Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 64 BYTE FRAME COUNTER [17:0] Incremented for each good or bad frame transmitted and received which is 64 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.10. TR511 (Ask a Question)

Name: TR511
Offset: 0x08C
Reset: 0x0
Property: Read-only

TR511 - Transmit & Receive 256 to 511 Byte Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 256 TO 511 BYTE FRAME COUNTER [17:0] Incremented for each good or bad frame transmitted and received which is 256 to 511 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.11. TR255 (Ask a Question)

Name: TR255
Offset: 0x088
Reset: 0x0
Property: Read-only

TR255 - Transmit & Receive 128 to 255 Byte Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 128 TO 255 BYTE FRAME COUNTER [17:0] Incremented for each good or bad frame transmitted and received which is 128 to 255 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.12. TR1K (Ask a Question)

Name: TR1K
Offset: 0x090
Reset: 0x0
Property: Read-only

TR1K - Transmit & Receive 512 to 1023 Byte Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 512 TO 1023 BYTE FRAME COUNTER [17:0] Incremented for each good or bad frame transmitted and received which is 512 to 1023 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.13. TR127 (Ask a Question)

Name: TR127
Offset: 0x084
Reset: 0x0
Property: Read-only

TR127 - Transmit & Receive 65 to 127 Byte Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT AND RECEIVE 65 TO 127 BYTE FRAME COUNTER [17:0] Incremented for each good or bad frame transmitted and received which is 65 to 127 bytes in length inclusive (excluding framing bits but including FCS bytes).

5.2.14. TPKT (Ask a Question)

Name: TPKT
Offset: 0x0E4
Reset: 0x0
Property: Read-only

TPKT- Transmit Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT PACKET COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT PACKET COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT PACKET COUNTER [17:0] Incremented for each transmitted packet (including bad packets, excessive deferred packets, excessive collision packets, late collision packets, all Unicast, Broadcast, and Multicast packets).

5.2.15. TOVR (Ask a Question)

Name: TOVR
Offset: 0x124
Reset: 0x0
Property: Read-only

TOVR- Transmit Oversize Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT OVERSIZE FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT OVERSIZE FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT OVERSIZE FRAME COUNTER [11:0] Incremented for each oversized transmitted frame with a correct FCS value.

5.2.16. TNCL (Ask a Question)

Name: TNCL
Offset: 0x10C
Reset: 0x0
Property: Read-only

TNCL- Transmit Total Collision Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT TOTAL COLLISION COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT TOTAL COLLISION COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT TOTAL COLLISION COUNTER [11:0] Incremented by the number of collisions experienced during the transmission of a frame as defined as the simultaneous presence of signals on the DO and RD circuits (that is transmitting and receiving at the same time).

5.2.17. TMCL (Ask a Question)

Name: TMCL
Offset: 0x100
Reset: 0x0
Property: Read-only

TMCL- Transmit Multiple Collision Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT MULTIPLE COLLISION PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT MULTIPLE COLLISION PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT MULTIPLE COLLISION PACKET COUNTER [11:0] Incremented for each frame transmitted which experienced 2-15 collisions (including any late collisions.

5.2.18. TMCA (Ask a Question)

Name: TMCA
Offset: 0x0E8
Reset: 0x0
Property: Read-only

TMCA- Transmit Multicast Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT MULTICAST PACKET COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT MULTICAST PACKET COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT MULTICAST PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT MULTICAST PACKET COUNTER [17:0] Incremented for each multicast valid frame transmitted (excluding Broadcast frames).

5.2.19. TLCL (Ask a Question)

Name: TLCL
Offset: 0x104
Reset: 0x0
Property: Read-only

TLCL- Transmit Late Collision Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT LATE COLLISION PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT LATE COLLISION PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT LATE COLLISION PACKET COUNTER [11:0] Incremented for each frame transmitted which experienced a late collision during a transmission attempt.

5.2.20. TJBR (Ask a Question)

Name: TJBR
Offset: 0x118
Reset: 0x0
Property: Read-only

TJBR-TransmitJabberFrameCounter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT JABBER FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT JABBER FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT JABBER FRAME COUNTER [11:0] Incremented for each oversized transmitted frame with an incorrect FCS value.

5.2.21. TFRG (Ask a Question)

Name: TFRG
Offset: 0x12C
Reset: 0x0
Property: Read-only

TFRG- Transmit Fragments Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT FRAGMENTS FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT FRAGMENTS FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT FRAGMENTS FRAME COUNTER [11:0] Incremented for every frame less than 64 bytes, with an incorrect FCS value.

5.2.22. TFCS (Ask a Question)

Name: TFCS
Offset: 0x11C
Reset: 0x0
Property: Read-only

TFCS- Transmit FCS Error Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT FCS ERROR COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT FCS ERROR COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT FCS ERROR COUNTER [11:0] Incremented for every valid sized packet with an incorrect FCS value.

5.2.23. Test_Register (Ask a Question)

Name: Test_Register
Offset: 0x01C
Reset: 0x0
Property: Read/Write

Test Register

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
					MAXIMUM BACKOFF	REGISTERED TRANSMIT FLOW ENABL	TEST PAUSE	SHORTCUT SLOT TIME
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 3 – MAXIMUM BACKOFF Setting this bit causes the MAC to back off for the maximum possible length of time. This test bit is used to predict back off times in Half_Duplex mode.

Bit 2 – REGISTERED TRANSMIT FLOW ENABL Transmit half_duplex Flow Enable

Bit 1 – TEST PAUSE Setting this bit allows the MAC to be paused through the APB target interface for testing purposes.

Bit 0 – SHORTCUT SLOT TIME Setting this bit allows the slot time counter to expire regardless of the current count. This bit is for testing purposes only. Upon PAUSE condition frame transmission gets paused until slot time counter reached h7e for 1G and h3e non 1G modes and it can be overcome by writing 1 to this bit.

5.2.24. TEDF (Ask a Question)

Name: TEDF
Offset: 0x0F8
Reset: 0x0
Property: Read-only

TEDF- Transmit Excessive Deferral Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT EXCESSIVE DEFERRAL PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT EXCESSIVE DEFERRAL PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT EXCESSIVE DEFERRAL PACKET COUNTER [11:0] Incremented for frames aborted which were deferred for an excessive period of time (3036 byte times).

5.2.25. TDRP (Ask a Question)

Name: TDRP
Offset: 0x114
Reset: 0x0
Property: Read-only

TDRP- Transmit Drop Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT DROP FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT DROP FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT DROP FRAME COUNTER [11:0] Incremented each time input PFH is asserted.

5.2.26. TDFR (Ask a Question)

Name: TDFR
Offset: 0x0F4
Reset: 0x0
Property: Read-only

TDFR- Transmit Deferral Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					TRANSMIT DEFERRAL PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT DEFERRAL PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – TRANSMIT DEFERRAL PACKET COUNTER [11:0] Incremented for each frame, which was deferred on its first transmission attempt. Does not include frames involved in collisions.

5.2.27. TDBEDC (Ask a Question)

Name: TDBEDC
Offset: 0x144
Reset: 0x0
Property: Read-only

TDBEDC Transmit ECC DED Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	TRANSMIT ECC DED COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT ECC DED COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – TRANSMIT ECC DED COUNTER [15:0] Incremented this counter whenever double bit error detect (DB_DETECT) flag of transmit buffer memory is asserted.

5.2.28. TBYT (Ask a Question)

Name: TBYT
Offset: 0x0E0
Reset: 0x0
Property: Read-only

TBYT- Transmit Byte Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	TRANSMIT BYTE COUNTER [23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT BYTE COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT BYTE COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 23:0 – TRANSMIT BYTE COUNTER [23:0] Incremented by the number of bytes that were put on the wire including fragments of frames that were involved with collisions. This count does not include preamble/SFD or jam bytes.

5.2.29. TBCA (Ask a Question)

Name: TBCA
Offset: 0x0EC
Reset: 0x0
Property: Read-only

TBCA- Transmit Broadcast Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							TRANSMIT BROADCAST PACKET COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	TRANSMIT BROADCAST PACKET COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TRANSMIT BROADCAST PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – TRANSMIT BROADCAST PACKET COUNTER [17:0] Incremented for each Broadcast frame transmitted (excluding Multicast frames).

5.2.30. Station_Address_Lower_Register (Ask a Question)

Name: Station_Address_Lower_Register
Offset: 0x040
Reset: 0x0
Property: Read/Write

Station Address Lower Register

Bit	31	30	29	28	27	26	25	24
	FIRST OCTET OF THE DA IN THE FRAME[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	SECOND OCTET OF THE DA IN THE FRAME[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	THIRD OCTET OF THE DA IN THE FRAME[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	FOURTH OCTET OF THE DA IN THE FRAME[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:24 – FIRST OCTET OF THE DA IN THE FRAME[7:0]

Bits 23:16 – SECOND OCTET OF THE DA IN THE FRAME[7:0]

Bits 15:8 – THIRD OCTET OF THE DA IN THE FRAME[7:0]

Bits 7:0 – FOURTH OCTET OF THE DA IN THE FRAME[7:0]

5.2.31. Station_Address_Higher_Register (Ask a Question)

Name: Station_Address_Higher_Register
Offset: 0x044
Reset: 0x0
Property: Read/Write

Station Address Higher Register

Bit	31	30	29	28	27	26	25	24
	FIFTH OCTET OF THE DA IN THE FRAME[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	SIXTH OCTET OF THE DA IN THE FRAME[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

Bits 31:24 – FIFTH OCTET OF THE DA IN THE FRAME[7:0]

Bits 23:16 – SIXTH OCTET OF THE DA IN THE FRAME[7:0]

5.2.32. RXUO (Ask a Question)

Name: RXUO
Offset: 0x0B8
Reset: 0x0
Property: Read-only

RXUO - Receive Unknown OPCode Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE UNKNOWN OPCODE PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE UNKNOWN OPCODE PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE UNKNOWN OPCODE PACKET COUNTER [11:0] Incremented each time a MAC Control Frame is received which contains an opcode other than a PAUSE.

5.2.33. RXPF (Ask a Question)

Name: RXPF
Offset: 0x0B4
Reset: 0x0
Property: Read-only

RXPF - Receive PAUSE Control Frame Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE PAUSE CONTROL FRAME COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE PAUSE CONTROL FRAME COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE PAUSE CONTROL FRAME COUNTER [11:0] Incremented each time a valid PAUSE MAC Control frame is received.

5.2.34. RXCF (Ask a Question)

Name: RXCF
Offset: 0x0B0
Reset: 0x0
Property: Read-only

RXCF - Receive Control Frame Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE CONTROL FRAME PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE CONTROL FRAME PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE CONTROL FRAME PACKET COUNTER [11:0] Incremented for each MAC Control frame received (PAUSE and Unsupported).

5.2.35. RUND (Ask a Question)

Name: RUND
Offset: 0x0CC
Reset: 0x0
Property: Read-only

RUND - Receive Undersize Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE UNDERSIZE PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE UNDERSIZE PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE UNDERSIZE PACKET COUNTER [11:0] Incremented each time a frame is received which is less than 64 bytes in length and contains a valid FCS and were otherwise well formed. This does not look at Range Length errors.

5.2.36. RSBECC (Ask a Question)

Name: RSBECC
Offset: 0x148
Reset: 0x0
Property: Read-only

RSBECC Receiver ECC SEC Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	RECEIVER ECC SEC COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVER ECC SEC COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – RECEIVER ECC SEC COUNTER [15:0] Incremented this counter whenever single bit error correct (SB_CORRECT) flag of receiver buffer memory is asserted.

5.2.37. RPKT (Ask a Question)

Name: RPKT
Offset: 0x0A0
Reset: 0x0
Property: Read-only

RPKT- Receive Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							RECEIVE PACKET COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	RECEIVE PACKET COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – RECEIVE PACKET COUNTER [17:0] Incremented for each frame received packet (including bad packets, all Unicast, Broadcast, and Multicast packets).

5.2.38. ROVR (Ask a Question)

Name: ROVR
Offset: 0x0D0
Reset: 0x0
Property: Read-only

ROVR - Receive Oversize Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE OVERSIZE PACKET COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE OVERSIZE PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE OVERSIZE PACKET COUNTER [11:0] Incremented each time a frame is received which exceeded 1518 (non VLAN) or 1522 (VLAN) and contains a valid FCS and were otherwise well formed. This does not look at Range Length errors.

5.2.39. RMCA (Ask a Question)

Name: RMCA
Offset: 0x0A8
Reset: 0x0
Property: Read-only

RMCA - Receive Multicast Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							RECEIVE MULTICAST PACKET COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	RECEIVE MULTICAST PACKET COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE MULTICAST PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – RECEIVE MULTICAST PACKET COUNTER [17:0] Incremented for each Multicast good frame of lengths smaller than 1518 (non VLAN) or 1522 (VLAN) excluding Broadcast frames. This does not look at range/length errors.

5.2.40. RJBR (Ask a Question)

Name: RJBR
Offset: 0x0D8
Reset: 0x0
Property: Read-only

RJBR - Receive Jabber Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE JABBER COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE JABBER COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE JABBER COUNTER [11:0] Incremented for frames received which exceed 1518 (non VLAN) or 1522 (VLAN) bytes and contains an invalid FCS, includes alignment errors.

5.2.41. RFRG (Ask a Question)

Name: RFRG
Offset: 0x0D4
Reset: 0x0
Property: Read-only

RFRG -Receive Fragments Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE FRAGMENTS COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE FRAGMENTS COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE FRAGMENTS COUNTER [11:0] Incremented for each frame received which is less than 64 bytes in length and contains an invalid FCS, includes integral and non-integral lengths.

5.2.42. RFLR (Ask a Question)

Name: RFLR
Offset: 0x0C0
Reset: 0x0
Property: Read-only

RFLR - Receive Frame Length Error Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE FRAME LENGTH ERROR COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE FRAME LENGTH ERROR COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE FRAME LENGTH ERROR COUNTER [11:0] Incremented for each frame received in which the 802.3 length field did not match the number of data bytes actually received (46-1500 bytes). The counter is not incremented if the length field is not a valid 802.3 length, such as an EtherType value.

5.2.43. RFCS (Ask a Question)

Name: RFCS
Offset: 0x0A4
Reset: 0x0
Property: Read-only

RFCS - Receive FCS Error Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE FCS ERROR COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE FCS ERROR COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE FCS ERROR COUNTER [11:0] Incremented for each frame received that has an integral 64 to 1518 length and contains a Frame Check Sequence error.

5.2.44. RDRP (Ask a Question)

Name: RDRP
Offset: 0x0DC
Reset: 0x0
Property: Read-only

RDRP - Receive Drop

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE DROP [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE DROP [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE DROP [11:0] Incremented for frames received which are streamed to system but are later dropped due to lack of system resources.

5.2.45. RDBEDC (Ask a Question)

Name: RDBEDC
Offset: 0x14C
Reset: 0x0
Property: Read-only

RDBEDC Receiver ECC DED counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	RECEIVER ECC DED COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVER ECC DED COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – RECEIVER ECC DED COUNTER [15:0] Incremented this counter whenever double bit error detect (DB_DETECT) flag of receiver buffer memory is asserted.

5.2.46. RCSE (Ask a Question)

Name: RCSE
Offset: 0x0C8
Reset: 0x0
Property: Read-only

RCSE - Receive Carrier Sense Error Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE CARRIER SENSE ERROR COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE CARRIER SENSE ERROR COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE CARRIER SENSE ERROR COUNTER [11:0] Incremented each time a false carrier is detected during idle, as defined by a 1 on RXER and an 0xE on RXD. The event is reported along with the statistics generated on the next received frame. Only one false carrier condition can be detected and logged between frames.

5.2.47. RCDE (Ask a Question)

Name: RCDE
Offset: 0x0C4
Reset: 0x0
Property: Read-only

RCDE- Receive Code Error Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE CODE ERROR COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE CODE ERROR COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE CODE ERROR COUNTER [11:0] Incremented each time a valid carrier was present and at least one invalid data symbol was detected.

5.2.48. RBYT (Ask a Question)

Name: RBYT
Offset: 0x09C
Reset: 0x0
Property: Read-only

RBYT - Receive Byte Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	RECEIVE BYTE COUNTER [23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	RECEIVE BYTE COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE BYTE COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 23:0 – RECEIVE BYTE COUNTER [23:0] The Statistic Counter register is incremented by the byte count of all frames received, including those in bad packets, excluding framing bits but including FCS bytes.

5.2.49. RBCA (Ask a Question)

Name: RBCA
Offset: 0x0AC
Reset: 0x0
Property: Read-only

RBCA - Receive Broadcast Packet Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							RECEIVE BROADCAST PACKET COUNTER [17:16]	
Access							R	R
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	RECEIVE BROADCAST PACKET COUNTER [15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE BROADCAST PACKET COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 17:0 – RECEIVE BROADCAST PACKET COUNTER [17:0] Incremented for each Broadcast good frame of lengths smaller than 1518 (non VLAN) or 1522 (VLAN) excluding Multicast frames. This does not look at range/length errors.

5.2.50. RALN (Ask a Question)

Name: RALN
Offset: 0x0BC
Reset: 0x0
Property: Read-only

RALN - Receive Alignment Error Counter

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RECEIVE ALIGNMENT ERROR COUNTER [11:8]			
Access					R	R	R	R
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RECEIVE ALIGNMENT ERROR COUNTER [7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 11:0 – RECEIVE ALIGNMENT ERROR COUNTER [11:0] Incremented for each received frame from 64 to 1518 which contains an invalid FCS and is not an integral number of bytes.

5.2.51. MIIMgmt_Status (Ask a Question)

Name: MIIMgmt_Status
Offset: 0x030
Reset: 0x0
Property: Read-only

MII Mgmt: Status

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	MDIO MGMT STATUS (PHY STATUS)[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	MDIO MGMT STATUS (PHY STATUS)[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – MDIO MGMT STATUS (PHY STATUS)[15:0] Following an MDIO Mgmt Read Cycle, the 16 bit data can be read from this location.

5.2.52. Misc_Control_register (Ask a Question)

Name: Misc_Control_register
Offset: 0x1D4
Reset: 0x0
Property: Read/Write

Misc Control register

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

5.2.53. MIIMgmt_Indicators (Ask a Question)

Name: MIIMgmt_Indicators
Offset: 0x034
Reset: 0x0
Property: Read-only

MII Mgmt: Indicators

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
						NOT VALID	SCANNING	BUSY
Access						R	R	R
Reset						0	0	0

Bit 2 – NOT VALID When 1 is returned indicates MDIO Mgmt Read cycle has not completed and the Read Data is not yet valid.

Bit 1 – SCANNING When 1 is returned-indicates a scan operation (continuous MDIO Mgmt Read cycles) is in progress.

Bit 0 – BUSY When 1 is returned-indicates MDIO Mgmt block is currently performing an MDIO Mgmt Read or Write cycle.

5.2.54. MIIMgmt_Control (Ask a Question)

Name: MIIMgmt_Control
Offset: 0x02C
Reset: 0x0
Property: Write-only

MII Mgmt: Control

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	MDIO MGMT CONTROL (PHY CONTROL)[15:8]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	MDIO MGMT CONTROL (PHY CONTROL)[7:0]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – MDIO MGMT CONTROL (PHY CONTROL)[15:0] When written, an MDIO Mgmt write cycle is performed using the 16-bit data and the preconfigured PHY and Register addresses from the MDIO Mgmt Address Register.

5.2.55. MIIMgmt_Configuration (Ask a Question)

Name: MIIMgmt_Configuration
Offset: 0x020
Reset: 0x0
Property: Read/Write

MII Mgmt: Configuration

Bit	31	30	29	28	27	26	25	24
	RESET MDIO MGMT							
Access	R/W							
Reset	0							

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
			SCAN AUTO INCREMENT	PREAMBLE SUPPRESSION		MGMT CLOCK SELECT[2:0]		
Access			R/W	R/W		R/W	R/W	R/W
Reset			0	0		0	0	0

Bit 31 – RESET MDIO MGMT Setting this bit resets MDIO Mgmt. Clearing this bit allows MDIO Mgmt to perform Mgmt read or write cycles as requested via the APB target interface.

Bit 5 – SCAN AUTO INCREMENT Setting this bit causes MDIO Mgmt to continually read from a set of PHYs of contiguous address space. The starting address of the PHY is specified by the content of the PHY address field recorded in the MDIO Mgmt Address register. The next PHY to be read is PHY address + 1. The last PHY to be queried in this read sequence is the one residing at address 0x31, after which the read sequence returns to the PHY specified by the PHY address field.

Bit 4 – PREAMBLE SUPPRESSION Setting this bit causes MDIO Mgmt to suppress preamble generation and reduce the Mgmt cycle from 64 clocks to 32 clocks. This is in accordance with IEEE 802.3/22.2.4.4.2. Clearing this bit causes MDIO Mgmt to perform Mgmt read/write cycles with the 64 clocks of preamble.

Bits 2:0 – MGMT CLOCK SELECT[2:0]

Value	Description
3b000/3b001	Source clock divided by 4
3b010	Source clock divided by 6
3b011	Source clock divided by 8
3b100	Source clock divided by 10
3b101	Source clock divided by 14
3b110	Source clock divided by 20
3b111	Source clock divided by 28

5.2.56. MIIMgmt_Command (Ask a Question)

Name: MIIMgmt_Command
Offset: 0x024
Reset: 0x0
Property: Read/Write

MII Mgmt: Command

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
							SCAN CYCLE	READ CYCLE
Access							R/W	R/W
Reset							0	0

Bit 1 – SCAN CYCLE This bit causes MDIO Mgmt to perform Read cycles continuously. This is useful for monitoring Link Fail.

Bit 0 – READ CYCLE This bit causes MDIO Mgmt to perform a single Read cycle. The Read data is returned in MDIO Mgmt STATUS Register.

5.2.57. MIIMgmt_Address (Ask a Question)

Name: MIIMgmt_Address
Offset: 0x028
Reset: 0x0
Property: Read/Write

MII Mgmt: Address

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
				PHY ADDRESS[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
				REGISTER ADDRESS[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bits 12:8 – PHY ADDRESS[4:0] This field represents the 5 bit PHY Address field used in Mgmt cycles. Up to 31 PHYs can be addressed.

Bits 4:0 – REGISTER ADDRESS[4:0] This field represents the 5 bit Register Address field of Mgmt cycles.

5.2.58. Maximum_Frame_Length (Ask a Question)

Name: Maximum_Frame_Length
Offset: 0x010
Reset: 0x7d0
Property: Read/Write

Maximum Frame Length

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	MAXIMUM FRAME LENGTH[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	1	1

Bit	7	6	5	4	3	2	1	0
	MAXIMUM FRAME LENGTH[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	0	1	0	0	0	0

Bits 15:0 – MAXIMUM FRAME LENGTH[15:0] This programmable field sets the maximum frame size in both the transmit and receive directions.The frame size includes the entire Layer 2 (L2) Ethernet frame, starting from the Destination MAC Address and ending with the Frame Check Sequence (FCS).

5.2.59. MACConfiguration2 (Ask a Question)

Name: MACConfiguration2
Offset: 0x004
Reset: 0x7200
Property: Read/Write

MAC Configuration # 2

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	PREAMBLE LENGTH[3:0]						INTERFACE MODE[1:0]	
Access	R/W	R/W	R/W	R/W			R/W	R/W
Reset	0	1	1	1			1	0

Bit	7	6	5	4	3	2	1	0
		RX CRC DISABLE	HUGE FRAME ENABLE	LENGTH FIELD CHECKING		PAD/CRC ENABLE	CRC ENABLE	FULL-DUPLEX
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 15:12 – PREAMBLE LENGTH[3:0] This field determines the length of the preamble field of the packet, in bytes. Minimum value supported for this field is 0x3.

Bits 9:8 – INTERFACE MODE[1:0]

Value	Description
2b00	MAC Tx/Rx represents MII 10Mbps interface (Nibble Mode)
2b01	MAC Tx/Rx represents MII 100Mbps interface (Nibble Mode)
2b10	MAC Tx/Rx represents GMII 1000Mbps interface (Byte Mode)
2b11	Reserved

Bit 6 – RX CRC DISABLE Setting this bit enables the removal of CRC bytes from all received frames at the MAC RX interface. Clearing this bit retains the CRC bytes at the end of each frame.

Bit 5 – HUGE FRAME ENABLE Setting this bit allows frames longer than the MAXIMUM FRAME LENGTH to be transmitted and received. Clear this bit to have the MAC limit the length of frames at the MAXIMUM FRAMELENGTH value (Maximum Frame Length is set in separate Maximum Frame Length register).

Bit 4 – LENGTH FIELD CHECKING Setting this bit causes the MAC to check the frame length field to ensure it matches the actual data field length. Clear this bit if no length field checking is desired.

Bit 2 – PAD/CRC ENABLE Set this bit to have the MAC pad all short frames and append a CRC to every frame whether or not padding was required. Clear this bit if frames presented to the MAC have a valid length and contain a CRC.

Bit 1 – CRC ENABLE Set this bit to have the MAC append a CRC to all frames. Clear this bit if frames presented to the MAC have a valid length and contain a valid CRC. If the PAD/CRC ENABLE Configuration bit or the per packet PAD/CRC ENABLE is set, CRC ENABLE is ignored.

Bit 0 – FULL-DUPLEX Setting this bit configures the MAC to operate in full duplex mode. Clearing this bit configures the MAC to operate in half duplex mode only.

5.2.60. MACConfiguration1 (Ask a Question)

Name: MACConfiguration1
Offset: 0x000
Reset: 0x80000000
Property: Read/Write

MAC Configuration # 1

Bit	31	30	29	28	27	26	25	24
	SOFT RESET	SIMULATION RESET						
Access	R/W	R/W						
Reset	1	0						

Bit	23	22	21	20	19	18	17	16
					RESET RX MAC CONTROL	RESET TX MAC CONTROL	RESET RX FUNCTION	RESET TX FUNCTION
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
								LOOP BACK
Access								R/W
Reset								0

Bit	7	6	5	4	3	2	1	0
			RECEIVE FLOW CONTROL ENABLE	TRANSMIT FLOW CONTROL ENABLE	SYNCHRONIZED RECEIVE ENABLE	RECEIVE ENABLE	SYNCHRONIZED TRANSMIT ENABLE	TRANSMIT ENABLE
Access			R/W	R/W	R	R/W	R	R/W
Reset			0	0	0	0	0	0

Bit 31 – SOFT RESET Setting this bit puts all modules within the MAC in reset except the APB target interface.

Bit 30 – SIMULATION RESET Setting this bit will reset those registers, such as the random backoff timer, which are not controlled by normal resets (simulation only).

Bit 19 – RESET RX MAC CONTROL Setting this bit will put the PERMC Receive MAC Control block in reset. This block detects Control frames and contains the pause timers.

Bit 18 – RESET TX MAC CONTROL Setting this bit will put the PETMC Transmit MAC Control block in reset. This block multiplexes data and Control frame transfers. It also responds to XOFF PAUSE Control frames.

Bit 17 – RESET RX FUNCTION Setting this bit will put the PERFN Receive Function block in reset. This block performs the receive frame protocol.

Bit 16 – RESET TX FUNCTION Setting this bit will put the PETFN Transmit Function block in reset. This block performs the frame transmission protocol.

Bit 8 – LOOP BACK Setting this bit causes the PETFN (Transmit Function) MAC Transmit outputs to be looped back to the MAC Receive inputs. Clearing this bit results in normal operation.

Bit 5 – RECEIVE FLOW CONTROL ENABLE Setting this bit causes the PERFN (Receive Function) Receive MAC Control to detect and act on PAUSE Flow Control frames. Clearing this bit causes the Receive MAC Control to ignore PAUSE Flow Control frames.

Bit 4 – TRANSMIT FLOW CONTROL ENABLE Setting this bit allows the PETMC Transmit MAC Control to send PAUSE Flow Control frames when requested by the system. Clearing this bit prevents the Transmit MAC Control from sending Flow Control frames.

Bit 3 – SYNCHRONIZED RECEIVE ENABLE Receive Enable synchronized to the receive stream.

Bit 2 – RECEIVE ENABLE Setting this bit allows the MAC to receive frames from the PHY. Clearing this bit prevents the reception of frames.

Bit 1 – SYNCHRONIZED TRANSMIT ENABLE Transmit Enable synchronized to the transmit stream.

Bit 0 – TRANSMIT ENABLE Setting this bit allows the MAC to transmit frames from the system. Clearing this bit prevents the transmission of frames.

5.2.61. IPG (Ask a Question)

Name: IPG
Offset: 0x008
Reset: 0x40605060
Property: Read/Write

IPG / IFG

Bit	31	30	29	28	27	26	25	24
		NON_BACK_TO_BACK INTER_PACKET_GAP PART1 (IPGR1)[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		1	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
		NON_BACK_TO_BACK INTER_PACKET_GAP PART2 (IPGR2)[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		1	1	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	MINIMUM IFG ENFORCEMENT[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	1	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		BACK_TO_BACK INTER_PACKET_GAP[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		1	1	0	0	0	0	0

Bits 30:24 – NON_BACK_TO_BACK INTER_PACKET_GAP PART1 (IPGR1)[6:0] This programmable field represents the optional carrier sense window referenced in IEEE 802.3/4.2.3.2.1 Carrier Deference. If a carrier is detected during the timing of IPGR1, the MAC defers to the carrier. If, however, the carrier becomes active after IPGR1, the MAC continues timing IPGR2 and transmits, knowingly causing a collision. This ensures fair access to the medium. The permitted range of values is 0x0 to IPGR2. Default is 0x40 (64d) which follows the two-thirds/one-thirds guideline.

Bits 22:16 – NON_BACK_TO_BACK INTER_PACKET_GAP PART2 (IPGR2)[6:0] This programmable field represents the Non-Back-to-Back Inter-Packet-Gap in bit times. Default is 0x60 (96d), which represents the minimum IPG of 96 bits.

Bits 15:8 – MINIMUM IFG ENFORCEMENT[7:0] Default 0x50. This programmable field represents the minimum size of IFG to enforce between frames(expressed in bit times). A frame whose IFG is less than that programmed is dropped. The default setting of 0x50 (80d) represents half of the nominal minimum IFG which is 160 bits.

Bits 6:0 – BACK_TO_BACK INTER_PACKET_GAP[6:0] Default 0x60. This programmable field represents the IPG between Back-to-Back packets (expressed in bit times). This is the IPG parameter used exclusively in full-duplex mode when two transmit packets are sent back-to-back. Set this field to the desired number of bits. The default setting of 0x60 (96d) represents the minimum IPG of 96 bits. Minimum value supported for this field is 0x28 (40d).

5.2.62. Interface_Status (Ask a Question)

Name: Interface_Status
Offset: 0x03C
Reset: 0x0
Property: Read-only

Interface Status

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
						WAKE ON LANE DETECTED	EXCESS DEFER	
Access						R	R	
Reset						0	0	

Bit	7	6	5	4	3	2	1	0
					LINK FAIL			
Access					R			
Reset					0			

Bit 10 – WAKE ON LANE DETECTED This bit is only used when the optional WoL module is integrated It is set when the MAC detects a Magic Packet and stays high until it is cleared by the assertion of Wake On Lane Detected Clear. Its reset value is low.

Bit 9 – EXCESS DEFER This bit sets when the MAC excessively defers a transmission. It clears when read. This bit latches high. Excessive Deferred is a condition when the MAC has deferred sending a packet for a time longer than the length of two maximum length frames.

Bit 3 – LINK FAIL When read as a 1, the MDIO management module has read the PHY link fail register to be 1. When read as a 0, the MDIO management module has read the PHY link fail register to be 0. Note that for asynchronous host accesses, this bit must be read at least once every scan read cycle of the PHY.

5.2.63. Interface_Control (Ask a Question)

Name: Interface_Control
Offset: 0x038
Reset: 0x0
Property: Read/Write

Interface Control

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
	WOL: UNICAST MATCH ENABLE	WOL: MAGIC PACKET DETECTION ENABLE	WOL: WAKE ON LANE DETECTED CLEAR STATUS CLEAR	STATS COUNTERS : AUTO CLEAR COUNTERS ON READ	STATS COUNTERS : CLEAR ALL COUNTERS	STATS COUNTERS : MODULE ENABLE		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0		

Bit 7 – WOL: UNICAST MATCH ENABLE Setting this bit configures WoL module to enable Wake On Lane Detected assertion based on Unicast match.

Bit 6 – WOL: MAGIC PACKET DETECTION ENABLE Setting this bit configures WoL module to enable Wake On Lane Detected assertion based on magic packet detection.

Bit 5 – WOL: WAKE ON LANE DETECTED CLEAR STATUS CLEAR When this bit is asserted, Wake On Lane Detected status is held low. When this bit is cleared, Wake On Lane Detected may become asserted appropriately.

Bit 4 – STATS COUNTERS : AUTO CLEAR COUNTERS ON READ Setting this bit enables auto-clear-on-read feature for all the counters.

Bit 3 – STATS COUNTERS : CLEAR ALL COUNTERS Setting this bit clears all the statistics counters.

Bit 2 – STATS COUNTERS : MODULE ENABLE Setting this bit enables statistics counter module.

5.2.64. Hash_Table_Register3 (Ask a Question)

Name: Hash_Table_Register3
Offset: 0x1D0
Reset: 0x0
Property: Read/Write

Hash Table Register3

Bit	31	30	29	28	27	26	25	24
	HASH TABLE REGISTER3 [31:24]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HASH TABLE REGISTER3 [23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HASH TABLE REGISTER3 [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HASH TABLE REGISTER3 [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HASH TABLE REGISTER3 [31:0] Hash Entries-[127:96].

5.2.65. Hash_Table_Register2 (Ask a Question)

Name: Hash_Table_Register2
Offset: 0x1CC
Reset: 0x0
Property: Read/Write

Hash Table Register2

Bit	31	30	29	28	27	26	25	24
	HASH TABLE REGISTER2 [31:24]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HASH TABLE REGISTER2 [23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HASH TABLE REGISTER2 [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HASH TABLE REGISTER2 [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HASH TABLE REGISTER2 [31:0] Hash Entries-[95:64]

5.2.66. Hash_Table_Register1 (Ask a Question)

Name: Hash_Table_Register1
Offset: 0x1C8
Reset: 0x0
Property: Read/Write

Hash Table Register1

Bit	31	30	29	28	27	26	25	24
	HASH TABLE REGISTER1 [31:24]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HASH TABLE REGISTER1 [23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HASH TABLE REGISTER1 [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HASH TABLE REGISTER1 [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HASH TABLE REGISTER1 [31:0] Hash Entries-[63:32]

5.2.67. Hash_Table_Register0 (Ask a Question)

Name: Hash_Table_Register0
Offset: 0x1C4
Reset: 0x0
Property: Read/Write

Hash Table Register0

Bit	31	30	29	28	27	26	25	24
	HASH TABLE REGISTER0 [31:24]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HASH TABLE REGISTER0 [23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HASH TABLE REGISTER0 [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HASH TABLE REGISTER0 [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HASH TABLE REGISTER0 [31:0] Hash Entries-[31:0]

5.2.68. Half-Duplex (Ask a Question)

Name: Half-Duplex
Offset: 0x00C
Reset: 0xa1f037
Property: Read/Write

Half-Duplex

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	ALTERNATE BINARY EXPONENTIAL BACKOFF TRUNCATION[3:0]				ALTERNATE BINARY EXPONENTIAL BACKOFF ENABLE	BACKPRESSURE NO BACKOFF	NO BACKOFF	EXCESSIVE DEFER
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	1	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8
	RETRANSMISSION MAXIMUM[3:0]						COLLISION WINDOW[9:8]	
Access	R/W	R/W	R/W	R/W			R/W	R/W
Reset	1	1	1	1			0	0

Bit	7	6	5	4	3	2	1	0
	COLLISION WINDOW[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	1	0	1	1	1

Bits 23:20 – ALTERNATE BINARY EXPONENTIAL BACKOFF TRUNCATION[3:0] This field is used when ALTERNATE BINARY EXPONENTIAL BACKOFF ENABLE is set. The value programmed is substituted for the Ethernet standard value of ten.

Bit 19 – ALTERNATE BINARY EXPONENTIAL BACKOFF ENABLE Setting this bit configures the Tx MAC to use the ALTERNATE BINARY EXPONENTIAL BACKOFF TRUNCATION setting instead of the 802.3 standard tenth collisions. The Standard specifies that any collision after the tenth uses 210-1 as the maximum backoff time. Clearing this bit causes the Tx MAC to follow the standard binary exponential backoff rule.

Bit 18 – BACKPRESSURE NO BACKOFF Setting this bit configures the Tx MAC to immediately re-transmit following a collision during back pressure operation. Clearing this bit causes the Tx MAC to follow the binary exponential back off rule.

Bit 17 – NO BACKOFF Setting this bit configures the Tx MAC to immediately re-transmit following a collision. Clearing this bit causes the Tx MAC to follow the binary exponential back off rule.

Bit 16 – EXCESSIVE DEFER Setting this bit configures the Tx MAC to allow the transmission of a packet that has been excessively deferred. Clearing this bit causes the Tx MAC to abort the transmission of a packet that has been excessively deferred.

Bits 15:12 – RETRANSMISSION MAXIMUM[3:0] This is a programmable field specifying the number of retransmission attempts following a collision before aborting the packet due to excessive collisions. The Standard specifies the maximum number of attempts to be 0xF (15d).

Bits 9:0 – COLLISION WINDOW[9:0] This programmable field represents the slot time or collision window during which collisions might occur in a properly configured network. Since the collision window starts at the beginning of transmission, the preamble and SFD are included. The default of 0x37 (55d) corresponds to the count of frame bytes at the end of the window.

5.2.69. Framefiltercontrols (Ask a Question)

Name: Framefiltercontrols
Offset: 0x1C0
Reset: 0x3f
Property: Read/Write

Frame filter controls

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
			FRAME FILTER CONTROLS[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			1	1	1	1	1	1

Bits 5:0 – FRAME FILTER CONTROLS[5:0]

Value	Description
[5]	Pass the frame if the hash table entry matches for Multicast-DA
[4]	Pass the frame if the hash table entry matches for Unicast-DA
[3]	Promiscuous mode, allow all the frames to pass
[2]	Pass the frame if its Unicast-DA matches the configured-DA
[1]	Pass all multicast frames
[0]	Pass all broadcast frames

5.2.70. DMATxStatus (Ask a Question)

Name: DMATxStatus
Offset: 0x188
Reset: 0x0
Property: Read/Write

Note: Transmit Status, Flags in this register are cleared by writing a one to the bit field.

Bit	31	30	29	28	27	26	25	24
							TX RAM SINGLE BIT ERROR DETECTED AND CORRECTED	TX RAM DOUBLE BIT ERROR DETECTED
Access							R/W	R/W
Reset							0	0

Bit	23	22	21	20	19	18	17	16
	TXPKTCOUNT[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
					TXBUSERROR		TXUNDERRUN	TXPKTSENT
Access					R/W		R/W	R/W
Reset					0		0	0

Bit 25 – TX RAM SINGLE BIT ERROR DETECTED AND CORRECTED This bit will be set when the sb_correct flag from the Tx ram gets asserted. When set, this bit indicates that a single bit error got detected and corrected during the valid data transfer.

Bit 24 – TX RAM DOUBLE BIT ERROR DETECTED This bit will be set when the db_detect flag from the Tx ram gets asserted. When set, this bit indicates that a double bit error got detected during the valid data transfer.

Bits 23:16 – TXPKTCOUNT[7:0] 8-bit transmit packet counter that is incremented whenever the built-in DMA controller successfully transfers a packet, and decremented whenever the host writes a 1 to bit 0 of this register.

Bit 3 – TXBUSERROR When set, indicates that a host Target split, retry or error response was received by the DMA controller.

Bit 1 – TXUNDERRUN Set whenever the DMA controller reads a set (logic 1) Empty Flag in the descriptor it is processing.

Bit 0 – TXPKTSENT When set, indicates that one or more packets have been successfully transferred. Writing a 1 to this bit reduces the TxPktCount value by one. The bit is cleared whenever TxPktCount is zero.

5.2.71. DMATxDescriptor (Ask a Question)

Name: DMATxDescriptor
Offset: 0x184
Reset: 0x0
Property: Read/Write

Pointer to Transmit Descriptor

Bit	31	30	29	28	27	26	25	24
	TOP 30 BITS OF DESCRIPTOR ADDRESS[29:22]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	TOP 30 BITS OF DESCRIPTOR ADDRESS[21:14]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	TOP 30 BITS OF DESCRIPTOR ADDRESS[13:6]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TOP 30 BITS OF DESCRIPTOR ADDRESS[5:0]						IGNORED[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:2 – TOP 30 BITS OF DESCRIPTOR ADDRESS[29:0] When TxEnable is set by the host, the built-in DMA controller reads this register to discover the location in host memory of the first transmit packet descriptor.

Bits 1:0 – IGNORED[1:0] Ignored by the DMA controller, since it is a requirement of the system that all descriptors are 32-bit aligned in host memory.

5.2.72. DMATxCtrl (Ask a Question)

Name: DMATxCtrl
Offset: 0x180
Reset: 0x100
Property: Read/Write

DMA Transmit Control

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	TX INTERRUPT COALESCING THRESHOLD COUNT[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	1

Bit	7	6	5	4	3	2	1	0
								TX ENABLE
Access								R/W
Reset								0

Bits 15:8 – TX INTERRUPT COALESCING THRESHOLD COUNT[7:0] This field is used for setting the Scatter-Gather-DMA interrupt coalescing threshold.

Bit 0 – TX ENABLE Setting this bit enables DMA transmit packet transfers. The bit is cleared by the built-in DMA controller whenever it encounters a Tx Underrun or Bus Error state.

5.2.73. DMARxStatus (Ask a Question)

Name: DMARxStatus
Offset: 0x194
Reset: 0x0
Property: Read/Write

Status of Rx Packet Transfers, Note- Flags in this register are cleared by writing a one to the bit field.

Bit	31	30	29	28	27	26	25	24
							RX RAM SINGLE BIT ERROR DETECTED AND CORRECTED	RX RAM DOUBLE BIT ERROR DETECTED
Access							R/W	R/W
Reset							0	0

Bit	23	22	21	20	19	18	17	16
	RXPKTCOUNT[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
					RXBUSERROR	RXOVERFLOW		RXPKTRECEIVED
Access					R/W	R/W		R/W
Reset					0	0		0

Bit 25 – RX RAM SINGLE BIT ERROR DETECTED AND CORRECTED This bit will be set when the sb_correct flag from the Rx ram gets asserted. When set, this bit indicates that a single bit error got detected and corrected during the valid data transfer.

Bit 24 – RX RAM DOUBLE BIT ERROR DETECTED This bit will be set when the db_detect flag from the Rx ram gets asserted. When set, this bit indicates that a double bit error got detected during the valid data transfer.

Bits 23:16 – RXPKTCOUNT[7:0] 8-bit receive packet counter that is incremented whenever the built-in DMA controller successfully transfers a packet, and is decremented whenever the host writes a 1 to bit zero of this register

Bit 3 – RXBUSERROR When set, indicates that a host Target split, retry or error response was received by the DMA controller.

Bit 2 – RXOVERFLOW Set whenever the DMA controller reads a Zero Empty Flag in the descriptor it is processing.

Bit 0 – RXPKTRECEIVED When set, indicates that one or more packets have been successfully transferred. Writing a 1 to this bit reduces the RxPktCount value by one. The bit is cleared whenever RxPktCount is zero.

5.2.74. DMARxDescriptor (Ask a Question)

Name: DMARxDescriptor
Offset: 0x190
Reset: 0x0
Property: Read/Write

Pointer to Receive Descriptor

Bit	31	30	29	28	27	26	25	24
	TOP 30 BITS OF DESCRIPTOR ADDRESS[29:22]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	TOP 30 BITS OF DESCRIPTOR ADDRESS[21:14]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	TOP 30 BITS OF DESCRIPTOR ADDRESS[13:6]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TOP 30 BITS OF DESCRIPTOR ADDRESS[5:0]						IGNORED[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:2 – TOP 30 BITS OF DESCRIPTOR ADDRESS[29:0] When RxEnable is set by the host, the built-in DMA controller reads this register to discover the location in host memory of the first receive packet descriptor.

Bits 1:0 – IGNORED[1:0] Ignored by the DMA controller, since it is a requirement of the system that all descriptors are 32-bit aligned in host memory.

5.2.75. DMARxCtrl (Ask a Question)

Name: DMARxCtrl
Offset: 0x18C
Reset: 0x100
Property: Read/Write

DMA Receive Control

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	RX INTERRUPT COALESCING THRESHOLD COUNT[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	1

Bit	7	6	5	4	3	2	1	0
								RX ENABLE
Access								R/W
Reset								0

Bits 15:8 – RX INTERRUPT COALESCING THRESHOLD COUNT[7:0] This field is used for setting the Scatter-Gather-DMA interrupt coalescing threshold.

Bit 0 – RX ENABLE Setting this bit enables DMA receive packet transfers. When set, the built-in DMA controller will start to receive a new packet whenever the FIFO indicates that a new packet is available (FRSOF asserted). The bit is cleared by the built-in DMA controller whenever it encounters an Rx Overflow or Bus Error state.

5.2.76. DMAIntrMask Interrupt Mask (Ask a Question)

Name: DMAIntrMask Interrupt Mask
Offset: 0x198
Reset: 0x0
Property: Read/Write

Setting a bit to 1 in this register enables the corresponding status signal as an interrupt source. The DMAInterrupt register below is the AND of all DMA status bits with this mask register.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
			TX_ECC_SEC INTERRUPT MASK	TX_ECC_DED INTERRUPT MASK	RX_ECC_SEC INTERRUPT MASK	RX_ECC_DED INTERRUPT MASK	RXPKT INTERRUPT COALESCING MASK	TXPKT INTERRUPT COALESCING MASK
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	BUS ERROR MASK	RX OVERFLOW MASK		RXPKTRECEIVED MASK	TX BUS ERROR MASK		TX UNDERRUN MASK	TXPKTSENT MASK
Access	R/W	R/W		R/W	R/W		R/W	R/W
Reset	0	0		0	0		0	0

Bit 13 – TX_ECC_SEC INTERRUPT MASK Setting this bit to 1 enables the generation of interrupt for a single bit error detected and corrected in Tx buffer during packet transfer.

Bit 12 – TX_ECC_DED INTERRUPT MASK Setting this bit to 1 enables generation of interrupt for a double bit error detected in Tx buffer during packet transfer.

Bit 11 – RX_ECC_SEC INTERRUPT MASK Setting this bit to 1 enables the generation of interrupt for a single bit error detected and corrected in Rx buffer during packet transfer.

Bit 10 – RX_ECC_DED INTERRUPT MASK Setting this bit to 1 enables the generation of interrupt for a double bit error detected in Rx buffer during packet transfer.

Bit 9 – RXPKT INTERRUPT COALESCING MASK Setting this bit to 1 enables the interrupt coalescing.

Bit 8 – TXPKT INTERRUPT COALESCING MASK Setting this bit to 1 enables the interrupt coalescing.

Bit 7 – BUS ERROR MASK Setting this bit to 1 enables the BusError bit in the DMARxStatus register as an interrupt source.

Bit 6 – RX OVERFLOW MASK Setting this bit to 1 enables the RxOverflow bit in the DMARxStatus register as an interrupt source.

Bit 4 – RXPKTRECEIVED MASK Setting this bit to 1 enables the RxPktReceived bit in the DMARxStatus register as an interrupt source.

Bit 3 – TX BUS ERROR MASK Setting this bit to 1 enables the BusError bit in the DMATxStatus register as an interrupt source.

Bit 1 – TX UNDERRUN MASK Setting this bit to 1 enables the TxUnderrun bit in the DMATxStatus register as an interrupt source.

Bit 0 – TXPKTSENT MASK Setting this bit to 1 enables the TxPktSent bit in the DMATxStatus register as an interrupt source.

5.2.77. DMAInterrupt Interrupt status (Ask a Question)

Name: DMAInterrupt Interrupt status
Offset: 0x19C
Reset: 0x0
Property: Read-only

Read-Only register. Flags in this register are cleared when the corresponding Status bit is cleared.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
			TX_ECC_SEC INTERRUPT STATUS	TX_ECC_DED INTERRUPT STATUS	RX_ECC_SEC INTERRUPT STATUS	RX_ECC_DED INTERRUPT STATUS	RXPKT INTERRUPT COALESCING	TXPKT INTERRUPT COALESCING
Access			R	R	R	R	R	R
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RX BUS ERROR	RX OVERFLOW		RXPKTRECEIVED	BUS ERROR		TX UNDERRUN	TXPKTSENT
Access	R	R		R	R		R	R
Reset	0	0		0	0		0	0

Bit 13 – TX_ECC_SEC INTERRUPT STATUS This interrupt will be asserted only when the corresponding Mask bit is enabled and sb_correct flag gets asserted from Tx ram during valid data transfer. Once asserted, it will be de-asserted only when read request comes for the respective counter (TSBECC).

Bit 12 – TX_ECC_DED INTERRUPT STATUS This interrupt will be asserted only when the corresponding Mask bit is enabled and db_detect flag gets asserted from Tx ram during valid data transfer. Once asserted, it will be de-asserted only when read request comes for the respective counter (TDBEDC).

Bit 11 – RX_ECC_SEC INTERRUPT STATUS This interrupt will be asserted only when the corresponding Mask bit is enabled and sb_correct flag gets asserted from Rx ram during valid data transfer. Once asserted, it will be de-asserted only when read request comes for the respective counter (RSBECC).

Bit 10 – RX_ECC_DED INTERRUPT STATUS This interrupt will be asserted only when the corresponding Mask bit is enabled and db_detect flag gets asserted from Rx ram during valid data transfer. Once asserted, it will be de-asserted only when read request comes for the respective counter (RDBEDC).

Bit 9 – RXPKT INTERRUPT COALESCING Set to 1 when number of RxPkt received count equals the interrupt coalescing Rx-Threshold count value. This information will result in interrupt only when the corresponding Mask bit is enabled. Once asserted, this interrupt will stay asserted till the RxPktCount field in RxStatus register has non-zero value.

Bit 8 – TXPKT INTERRUPT COALESCING Set to 1 when number of TxPkt transmitted count equals the interrupt coalescing Tx-Threshold count value. This information will result in interrupt only when the

corresponding Mask bit is enabled. Once asserted, this interrupt will stay asserted till the TxPktCount field in TxStatus register has non-zero value.

Bit 7 – RX BUS ERROR Set to 1 to record a Receive Bus Error interrupt when the BusError bit in the DMARxStatus register and bit 7 of the DMAIntrMask register are both set.

Bit 6 – RX OVERFLOW Set to 1 to record an Rx Overflow interrupt when the RxOverflow bit in the DMARxStatus register and bit 6 of the DMAIntrMask register are both set.

Bit 4 – RXPKTRECEIVED Set to 1 to record a RxPkt Received interrupt when the RxPktReceived bit in the DMARxStatus register and bit 4 of the DMAIntrMask register are both set.

Bit 3 – BUS ERROR Set to 1 to record a Transmit BusError interrupt when the BusError bit in the DMATxStatus register and bit 3 of the DMAIntrMask register are both set.

Bit 1 – TX UNDERRUN Set to 1 to record a Tx Underrun interrupt when the TxUnderrun bit in the DMATxStatus register and bit 1 of the DMAIntrMask register are both set.

Bit 0 – TXPKTSENT Set to 1 to record a Tx Pkt Sent interrupt when the TxPktSent bit in the DMATxStatus register and bit 0 of the DMAIntrMask register are both set.

5.2.78. Control_Frame_parameter (Ask a Question)

Name: Control_Frame_parameter
Offset: 0x018
Reset: 0xffff
Property: Read/Write

Control Frame parameter (Used for pause Value)

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	CFPT[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	CFPT[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bits 15:0 – CFPT[15:0] This register bits are append as Pause frame payload.

5.2.79. Control_Frame_extended_parameter (Ask a Question)

Name: Control_Frame_extended_parameter
Offset: 0x014
Reset: 0x0
Property: Read/Write

Control Frame extended parameter (Used for pause frame)

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	CFEP[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	CFEP[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – CFEP[15:0] This register bits are append as Pause frame payload.

5.2.80. CAR2 (Ask a Question)

Name: CAR2
Offset: 0x134
Reset: 0x0
Property: Read/Write-1-to Clear

CAR2 - Carry Register Two Register. This indicates the transmit counters carry bits. The carry register bits are cleared on carry register write when the respective bit is asserted.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			CARRY REGISTER 2 TSBECC COUNTER CARRY BIT	CARRY REGISTER 2 TDBEDC COUNTER CARRY BIT	CARRY REGISTER 2 TJBR COUNTER CARRY BIT	CARRY REGISTER 2 TXFC COUNTER CARRY BIT	CARRY REGISTER 2 TXCF COUNTER CARRY BIT	CARRY REGISTER 2 TOVR COUNTER CARRY BIT
Access			R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	CARRY REGISTER 2 TUND COUNTER CARRY BIT	CARRY REGISTER 2 TFRG COUNTER CARRY BIT	CARRY REGISTER 2 TBYT COUNTER CARRY BIT	CARRY REGISTER 2 TPKT COUNTER CARRY BIT	CARRY REGISTER 2 TMCA COUNTER CARRY BIT	CARRY REGISTER 2 TBCA COUNTER CARRY BIT	CARRY REGISTER 2 TXPF COUNTER CARRY BIT	CARRY REGISTER 2 TDFR COUNTER CARRY BIT
Access	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	CARRY REGISTER 2 TEDF COUNTER CARRY BIT	CARRY REGISTER 2 TSCL COUNTER CARRY BIT	CARRY REGISTER 2 TMCL COUNTER CARRY BIT	CARRY REGISTER 2 TLCL COUNTER CARRY BIT	CARRY REGISTER 2 TXCL COUNTER CARRY BIT	CARRY REGISTER 2 TNCL COUNTER CARRY BIT	CARRY REGISTER 2 TPFH COUNTER CARRY BIT	CARRY REGISTER 2 TDRP COUNTER CARRY BIT
Access	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C
Reset	0	0	0	0	0	0	0	0

Bit 21 – CARRY REGISTER 2 TSBECC COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Single Bit Error Correct Counter.

Bit 20 – CARRY REGISTER 2 TDBEDC COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Double Bit Error Detect Counter.

Bit 19 – CARRY REGISTER 2 TJBR COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Jabber Frame Counter.

Bit 18 – CARRY REGISTER 2 TXFC COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit FCS Error Counter.

Bit 17 – CARRY REGISTER 2 TXCF COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Control Frame Counter.

Bit 16 – CARRY REGISTER 2 TOVR COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Oversize Frame Counter.

Bit 15 – CARRY REGISTER 2 TUND COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Under size Frame Counter.

Bit 14 – CARRY REGISTER 2 TFRG COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Fragments Frame Counter.

Bit 13 – CARRY REGISTER 2 TBYT COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Byte Counter.

Bit 12 – CARRY REGISTER 2 TPKT COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Packet Counter.

Bit 11 – CARRY REGISTER 2 TMCA COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Multicast Packet Counter.

Bit 10 – CARRY REGISTER 2 TBCA COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Broadcast Packet Counter.

Bit 9 – CARRY REGISTER 2 TXPF COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit PAUSE Control Frame Counter.

Bit 8 – CARRY REGISTER 2 TDFR COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Deferral Packet Counter.

Bit 7 – CARRY REGISTER 2 TEDF COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Excessive Deferral Packet Counter.

Bit 6 – CARRY REGISTER 2 TSCL COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Single Collision Packet Counter.

Bit 5 – CARRY REGISTER 2 TMCL COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Multiple Collision Packet Counter.

Bit 4 – CARRY REGISTER 2 TLCL COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Late Collision Packet Counter.

Bit 3 – CARRY REGISTER 2 TXCL COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Excessive Collision Packet Counter.

Bit 2 – CARRY REGISTER 2 TNCL COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Total Collision Counter.

Bit 1 – CARRY REGISTER 2 TPFH COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Pause Frames Honored Counter.

Bit 0 – CARRY REGISTER 2 TDRP COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit Drop Frame Counter.

5.2.81. CAR1 (Ask a Question)

Name: CAR1
Offset: 0x130
Reset: 0x0
Property: Read/Write-1-to Clear

CAR1 - Carry Register One. This register indicates the transmit and receive counters carry bits. The carry registers bits are cleared on carry register write when the respective bit is asserted.

Bit	31	30	29	28	27	26	25	24
	CARRY REGISTER 1 TR64 COUNTER CARRY BIT	CARRY REGISTER 1 TR127 COUNTER CARRY BIT	CARRY REGISTER 1 TR255 COUNTER CARRY BIT	CARRY REGISTER 1 TR511 COUNTER CARRY BIT	CARRY REGISTER 1 TR1K COUNTER CARRY BIT	CARRY REGISTER 1 TRMAX COUNTER CARRY BIT	CARRY REGISTER 1 TRMGV COUNTER CARRY BIT	
Access	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	
Reset	0	0	0	0	0	0	0	

Bit	23	22	21	20	19	18	17	16
						CARRY REGISTER 1 RSBECC COUNTER CARRY BIT	CARRY REGISTER 1 RDBEDC COUNTER CARRY BIT	CARRY REGISTER 1 RBYT COUNTER CARRY BIT
Access						R/W1C	R/W1C	R/W1C
Reset						0	0	0

Bit	15	14	13	12	11	10	9	8
	CARRY REGISTER 1 RPKT COUNTER CARRY BIT	CARRY REGISTER 1 RFCS COUNTER CARRY BIT	CARRY REGISTER 1 RMCA COUNTER CARRY BIT	CARRY REGISTER 1 RBCA COUNTER CARRY BIT	CARRY REGISTER 1 RXCF COUNTER CARRY BIT	CARRY REGISTER 1 RXPF COUNTER CARRY BIT	CARRY REGISTER 1 RXUO COUNTER CARRY BIT	CARRY REGISTER 1 RALN COUNTER CARRY BIT
Access	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	CARRY REGISTER 1 RFLR COUNTER CARRY BIT	CARRY REGISTER 1 RCDE COUNTER CARRY BIT	CARRY REGISTER 1 RCSE COUNTER CARRY BIT	CARRY REGISTER 1 RUND COUNTER CARRY BIT	CARRY REGISTER 1 ROVR COUNTER CARRY BIT	CARRY REGISTER 1 RFRG COUNTER CARRY BIT	CARRY REGISTER 1 RJBR COUNTER CARRY BIT	CARRY REGISTER 1 RDRP COUNTER CARRY BIT
Access	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C	R/W1C
Reset	0	0	0	0	0	0	0	0

Bit 31 – CARRY REGISTER 1 TR64 COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 64 Byte Frame Counter.

Bit 30 – CARRY REGISTER 1 TR127 COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 65 to 127 Byte Frame Counter.

Bit 29 – CARRY REGISTER 1 TR255 COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 128 to 255 Byte Frame Counter.

Bit 28 – CARRY REGISTER 1 TR511 COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 256 to 511 Byte Frame Counter

Bit 27 – CARRY REGISTER 1 TR1K COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 512 to 1023 Byte Frame Counter.

Bit 26 – CARRY REGISTER 1 TRMAX COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 1024 to 1518 Byte Frame Counter.

Bit 25 – CARRY REGISTER 1 TRMGV COUNTER CARRY BIT Indicates the status of the Carry bit of Transmit and Receive 1519 to 1522 Byte VLAN Frame Counter.

Bit 18 – CARRY REGISTER 1 RSBECC COUNTER CARRY BIT Indicates the status of the Carry bit of Receiver Single Bit Error Correct Counter.

Bit 17 – CARRY REGISTER 1 RDBEDC COUNTER CARRY BIT Indicates the status of the Carry bit of Receiver Double Bit Error Detect Counter.

Bit 16 – CARRY REGISTER 1 RBYT COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Byte Counter.

Bit 15 – CARRY REGISTER 1 RPKT COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Packet Counter.

Bit 14 – CARRY REGISTER 1 RFCS COUNTER CARRY BIT Indicates the status of the Carry bit of Receive FCS Error Counter.

Bit 13 – CARRY REGISTER 1 RMCA COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Multicast Packet Counter.

Bit 12 – CARRY REGISTER 1 RBCA COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Broadcast Packet Counter.

Bit 11 – CARRY REGISTER 1 RXCF COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Control Frame Packet Counter.

Bit 10 – CARRY REGISTER 1 RXPF COUNTER CARRY BIT Indicates the status of the Carry bit of Receive PAUSE Control Frame Counter.

Bit 9 – CARRY REGISTER 1 RXUO COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Unknown OP code Packet Counter.

Bit 8 – CARRY REGISTER 1 RALN COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Alignment Error Counter.

Bit 7 – CARRY REGISTER 1 RFLR COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Frame Length Error Counter.

Bit 6 – CARRY REGISTER 1 RCDE COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Code Error Counter.

Bit 5 – CARRY REGISTER 1 RCSE COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Carrier Sense Error Counter.

Bit 4 – CARRY REGISTER 1 RUND COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Undersize Packet Counter.

Bit 3 – CARRY REGISTER 1 ROVR COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Oversize Packet Counter.

Bit 2 – CARRY REGISTER 1 RFRG COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Fragments Counter.

Bit 1 – CARRY REGISTER 1 RJBR COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Jabber Counter Receive Jabber Counter.

Bit 0 – CARRY REGISTER 1 RDRP COUNTER CARRY BIT Indicates the status of the Carry bit of Receive Drop Counter.

5.2.82. CAM2 (Ask a Question)

Name: CAM2
Offset: 0x13C
Reset: 0x3fffff
Property: Read/Write

CAM2 - Carry Register Two Mask Register. The rollover condition of each transmit counter can be discreetly masked from causing an interrupt by internal masking.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			MASK REGISTER 2 TSBECC COUNTER CARRY BIT	MASK REGISTER 2 TDBEDC COUNTER CARRY BIT	MASK REGISTER 2 TJBR COUNTER CARRY BIT	MASK REGISTER 2 TXFC COUNTER CARRY BIT	MASK REGISTER 2 TXCF COUNTER CARRY BIT	MASK REGISTER 2 TOVR COUNTER CARRY BIT
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8
	MASK REGISTER 2 TUND COUNTER CARRY BIT	MASK REGISTER 2 TFRG COUNTER CARRY BIT	MASK REGISTER 2 TBYT COUNTER CARRY BIT	MASK REGISTER 2 TPKT COUNTER CARRY BIT	MASK REGISTER 2 TMCA COUNTER CARRY BIT	MASK REGISTER 2 TBCA COUNTER CARRY BIT	MASK REGISTER 2 TXPF COUNTER CARRY BIT	MASK REGISTER 2 TDFR COUNTER CARRY BIT
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	MASK REGISTER 2 TEDF COUNTER CARRY BIT	MASK REGISTER 2 TSCL COUNTER CARRY BIT	MASK REGISTER 2 TMCL COUNTER CARRY BIT	MASK REGISTER 2 TLCL COUNTER CARRY BIT	MASK REGISTER 2 TXCL COUNTER CARRY BIT	MASK REGISTER 2 TNCL COUNTER CARRY BIT	MASK REGISTER 2 TPFH COUNTER CARRY BIT	MASK REGISTER 2 TDRP COUNTER CARRY BIT
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit 21 – MASK REGISTER 2 TSBECC COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Single Bit Error Correct Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 20 – MASK REGISTER 2 TDBEDC COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Double Bit Error Detect Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 19 – MASK REGISTER 2 TJBR COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Jabber Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 18 – MASK REGISTER 2 TXFC COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit FCS Error Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 17 – MASK REGISTER 2 TXCF COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Control Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 16 – MASK REGISTER 2 TOVR COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Oversize Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 15 – MASK REGISTER 2 TUND COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Under size Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 14 – MASK REGISTER 2 TFRG COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Fragments Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 13 – MASK REGISTER 2 TBYT COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Byte Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 12 – MASK REGISTER 2 TPKT COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 11 – MASK REGISTER 2 TMCA COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Multicast Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 10 – MASK REGISTER 2 TBCA COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Broadcast Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 9 – MASK REGISTER 2 TXPF COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit PAUSE Control Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 8 – MASK REGISTER 2 TDFR COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Deferral Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 7 – MASK REGISTER 2 TEDF COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Excessive Deferral Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 6 – MASK REGISTER 2 TSCL COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Single Collision Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 5 – MASK REGISTER 2 TMCL COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Multiple Collision Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 4 – MASK REGISTER 2 TLCL COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Late Collision Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 3 – MASK REGISTER 2 TXCL COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Excessive Collision Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 2 – MASK REGISTER 2 TNCL COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Total Collision Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 1 – MASK REGISTER 2 TPFH COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Pause Frames Honored Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 0 – MASK REGISTER 2 TDRP COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit Drop Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

5.2.83. CAM1 (Ask a Question)

Name: CAM1
Offset: 0x138
Reset: 0xFE07FFFF
Property: Read/Write

CAM1 - Carry Register One Mask Register. The rollover condition of each transmits and receive counter and receive counters can be discreetly masked from causing an interrupt by internal masking.

Bit	31	30	29	28	27	26	25	24
	MASK REGISTER 1 TR64 COUNTER CARRY BIT	MASK REGISTER 1 TR127 COUNTER CARRY BIT	MASK REGISTER 1 TR255 COUNTER CARRY BIT	MASK REGISTER 1 TR511 COUNTER CARRY BIT	MASK REGISTER 1 TR1K COUNTER CARRY BIT	MASK REGISTER 1 TRMAX COUNTER CARRY BIT	MASK REGISTER 1 TRMGV COUNTER CARRY BIT	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	1	1	1	1	1	1	1	

Bit	23	22	21	20	19	18	17	16
						MASK REGISTER 1 RSBECC COUNTER CARRY BIT	MASK REGISTER 1 RDBEDC COUNTER CARRY BIT	MASK REGISTER 1 RBYT COUNTER CARRY BIT
Access						R/W	R/W	R/W
Reset						1	1	1

Bit	15	14	13	12	11	10	9	8
	MASK REGISTER 1 RPKT COUNTER CARRY BIT	MASK REGISTER 1 RFCS COUNTER CARRY BIT	MASK REGISTER 1 RMCA COUNTER CARRY BIT	MASK REGISTER 1 RBCA COUNTER CARRY BIT	MASK REGISTER 1 RXCF COUNTER CARRY BIT	MASK REGISTER 1 RXPF COUNTER CARRY BIT	MASK REGISTER 1 RXUO COUNTER CARRY BIT	MASK REGISTER 1 RALN COUNTER CARRY BIT
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	MASK REGISTER 1 RFLR COUNTER CARRY BIT	MASK REGISTER 1 RCDE COUNTER CARRY BIT	MASK REGISTER 1 RCSE COUNTER CARRY BIT	MASK REGISTER 1 RUND COUNTER CARRY BIT	MASK REGISTER 1 ROVR COUNTER CARRY BIT	MASK REGISTER 1 RFRG COUNTER CARRY BIT	MASK REGISTER 1 RJBR COUNTER CARRY BIT	MASK REGISTER 1 RDRP COUNTER CARRY BIT
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit 31 – MASK REGISTER 1 TR64 COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 64 Byte Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 30 – MASK REGISTER 1 TR127 COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 65 to 127 Byte Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 29 – MASK REGISTER 1 TR255 COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 128 to 255 Byte Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 28 – MASK REGISTER 1 TR511 COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 256 to 511 Byte Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 27 – MASK REGISTER 1 TR1K COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 512 to 1023 Byte Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 26 – MASK REGISTER 1 TRMAX COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 1024 to 1518 Byte Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 25 – MASK REGISTER 1 TRMGV COUNTER CARRY BIT Mask or Unmask the carry bit of Transmit and Receive 1519 to 1522 Byte VLAN Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 18 – MASK REGISTER 1 RSBECC COUNTER CARRY BIT Mask or Unmask the carry bit of Receiver Single Bit Error Correct Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 17 – MASK REGISTER 1 RDBEDC COUNTER CARRY BIT Mask or Unmask the carry bit of Receiver Double Bit Error Detect counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 16 – MASK REGISTER 1 RBYT COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Byte Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 15 – MASK REGISTER 1 RPKT COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 14 – MASK REGISTER 1 RFCS COUNTER CARRY BIT Mask or Unmask the carry bit of Receive FCS Error Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 13 – MASK REGISTER 1 RMCA COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Multicast Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bi

Bit 12 – MASK REGISTER 1 RBCA COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Broadcast Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 11 – MASK REGISTER 1 RXCF COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Control Frame Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 10 – MASK REGISTER 1 RXPF COUNTER CARRY BIT Mask or Unmask the carry bit of Receive PAUSE Control Frame Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 9 – MASK REGISTER 1 RXUO COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Unknown OP code Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 8 – MASK REGISTER 1 RALN COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Alignment Error Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 7 – MASK REGISTER 1 RFLR COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Frame Length Error Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 6 – MASK REGISTER 1 RCDE COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Code Error Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 5 – MASK REGISTER 1 RCSE COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Carrier Sense Error Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 4 – MASK REGISTER 1 RUND COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Undersize Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 3 – MASK REGISTER 1 ROVR COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Oversize Packet Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 2 – MASK REGISTER 1 RFRG COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Fragments Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 1 – MASK REGISTER 1 RJBR COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Jabber Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

Bit 0 – MASK REGISTER 1 RDRP COUNTER CARRY BIT Mask or Unmask the carry bit of Receive Drop Counter.

Value	Description
0	Unmask the counter carry bit
1	Mask the counter carry bit

5.2.84. A-MCXFIFRAMAccess7 (Ask a Question)

Name: A-MCXFIFRAMAccess7
Offset: 0x07C
Reset: 0x0
Property: Read-only

A-MCXFIF FIFO RAM Access* Register 7

Bit	31	30	29	28	27	26	25	24
	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[31:0]) Host receive RAM read data. This is the lower 4 bytes of receive FIFO RAM data that is read at the address of hstrramradx[RABITS:0] if hstrramradx[RABITS+2] is negated and hstrramreq is asserted. This is a read-only field. Writes specifically to this field have no effects.

5.2.85. A-MCXFIFRAMAccess6 (Ask a Question)

Name: A-MCXFIFRAMAccess6
Offset: 0x078
Reset: 0x0
Property: Read/Write

A-MCXFIF FIFO RAM Access* Register 6

Bit	31	30	29	28	27	26	25	24
	HOST RECEIVE RAM READ REQUEST (HSTRRAMRREQ)	HOST RECEIVE RAM READ ACKNOWLEDGE (HSTRRAMRACK)						
Access	R/W	R/W						
Reset	0	0						

Bit	23	22	21	20	19	18	17	16
	HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[39:32])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			HOST RECEIVE RAM READ ADDRESS (HSTRRAMRADX[31:0])					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST RECEIVE RAM READ ADDRESS (HSTRRAMRADX[31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 31 – HOST RECEIVE RAM READ REQUEST (HSTRRAMRREQ) Host receive RAM read request. Requests the handshake of hstrramradx values to the receive FIFO RAM and hstrramrdat from the receive FIFO RAM. Should only be asserted while hstrramrack is negated and while the receive data path isdisabled from receiving data and is in a steady state. It should only be negated after receiving an asserted hstrramrack.

Bit 30 – HOST RECEIVE RAM READ ACKNOWLEDGE (HSTRRAMRACK) Host receive RAM read acknowledge. Signifies the acceptance of hstrramradx values to the receive FIFO RAM and reception of hstrramrdat from the receive FIFO RAM location addressed. Will only be asserted or negated following assertion or negation of hstrramrreq. This is a readonly bit. Writes specifically to this bit have no effects.

Bits 23:16 – HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[39:32]) Host receive RAM read data. This is the upper byte of receive FIFO RAM data that was read at the address of hstrramwadx[RABITS:0] if hstrramwadx[RABITS+2] is negated and hstrramwreq is asserted. This part of the receive FIFO RAM contains control information for the frame as follows:

Value	Description
hstrramwdat [39:36]	Unused
hstrramwdat [35]	System receive start of frame
hstrramwdat [34]	System receive end of frame

HOST RECEIVE RAM READ DATA (HSTRRAMRDAT[39:32]) (continued)	
Value	**Description**
hstrramwdat [33:32]	Data valid byte enable and applicable only for the last word of the frame.
hstrramwdat [33:32] = 0	Indicates all bytes in the word are valid
hstrramwdat [33:32] = 1	Indicates the LSB 3 bytes are valid [23:0] bits
hstrramwdat [33:32] = 2	Indicates the LSB 2 bytes are valid [15:0] bits
hstrramwdat [33:32] = 3	Indicates the LSB 1 bytes are valid [7:0] bits

Bits 13:0 – HOST RECEIVE RAM READ ADDRESS (HSTRRAMRADX[31:0]) Host receive RAM read address. If hstrramradx[RABITS+2] is written low, hstrramradx[11:0] is the receive FIFO RAM address which hstrramrdat is read from. If hstrramradx[RABITS+2] is written high, hstrramradx[12:0] contains the pointer value read from abric receive module.

5.2.86. A-MCXFIFRAMAccess5 (Ask a Question)

Name: A-MCXFIFRAMAccess5
Offset: 0x074
Reset: 0x0
Property: Read/Write

A-MCXFIF FIFO RAM Access* Register 5

Bit	31	30	29	28	27	26	25	24
	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT [31:0]) Host receive RAM write data. This is the lower 4 bytes of receive FIFO RAM data that writes at the address of hstrramwadx[RABITS:0] if hsttramwadx[RABITS+2] is negated and hstrramwreq is asserted.

5.2.87. A-MCXFIFRAMAccess4 (Ask a Question)

Name: A-MCXFIFRAMAccess4
Offset: 0x070
Reset: 0x0
Property: Read/Write

A-MCXFIF FIFO RAM Access* Register 4

Bit	31	30	29	28	27	26	25	24
	HOST RECEIVE RAM WRITE REQUEST (HSTRRAMWREQ)	HOST RECEIVE RAM WRITE ACKNOWLEDGE (HSTRRAMWACK)						
Access	R/W	R						
Reset	0	0						

Bit	23	22	21	20	19	18	17	16
	HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT[39:32])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		HOST RECEIVE RAM WRITE ADDRESS (HSTTRAMWADX[(RABITS+2):0])						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST RECEIVE RAM WRITE ADDRESS (HSTTRAMWADX[(RABITS+2):0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 31 – HOST RECEIVE RAM WRITE REQUEST (HSTRRAMWREQ) Host receive RAM write request. Requests the handshake of hstrramwdat and hstrramwadx values to the receive FIFO RAM. Only be asserted while hstrramwack is negated and while the receive data path is disabled from receiving data in a steady state.

Bit 30 – HOST RECEIVE RAM WRITE ACKNOWLEDGE (HSTRRAMWACK) Host receive RAM write acknowledge. Signifies the acceptance of hstrramwdat and hstrramwadx values to the receive FIFO RAM or System receive module. Will only be asserted or negated following assertion or negation of hstrramwreq. This is a read-only bit. Writes specifically to this bit no effects.

Bits 23:16 – HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT[39:32]) Host receive RAM write data. This is the upper byte of receive FIFO RAM data that is written at the address of hstrramwadx[RABITS:0] if hstrramwadx[RABITS+2] is negated and hstrramwreq is asserted. This part of the receive FIFO RAM contains control information for the frame as follows:

Value	Description
hstrramwdat [39:36]	Unused
hstrramwdat [35]	System receive start of frame
hstrramwdat [34]	System receive end of frame
hstrramwdat [33:32]	Data valid byte enable and applicable only for the last word of the frame.

HOST RECEIVE RAM WRITE DATA (HSTRRAMWDAT[39:32]) (continued)	
Value	**Description**
hstrramwdat [33:32] = 0	Indicates all bytes in the word are valid
hstrramwdat [33:32] = 1	Indicates the LSB 3 bytes are valid [23:0] bits
hstrramwdat [33:32] = 2	Indicates the LSB 2 bytes are valid [15:0] bits
hstrramwdat [33:32] = 3	Indicates the LSB 1 bytes are valid [7:0] bits

Bits 14:0 – HOST RECEIVE RAM WRITE ADDRESS (HSTTRAMWADX[(RABITS+2):0]) Host receive RAM write address. This field has different functionality based on the value of hstrramwadx[RABITS+1] and whether it is being written to or read from. When read from, hstrramwadx[12:0] field contains the actual write pointer value of the System receieve module. When written to the hstrramwadx register is loaded. If hstrramwadx[RABITS+1] is low, hstrramwadx[11:0] receives FIFO RAM address which hstrramwdat is written to. If hstrramwadx[RABITS+2] is high, hstrramwadx[RABITS+1:0] contains the pointer value that is written to System receieve module. The bit field length shown corresponds to a receive RAM configured with 12 address bits (8 KB). The actual bit field width varies based on the configured receive RAM size. The upper limit of this bit-field is derived from the RABITS parameter with the effective range as [(RABITS + 2 : 0].

5.2.88. A-MCXFIFRAMAccess3 (Ask a Question)

Name: A-MCXFIFRAMAccess3
Offset: 0x06C
Reset: 0x0
Property: Read-only

A-MCXFIF FIFO RAM Access* Register 3

Bit	31	30	29	28	27	26	25	24
	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[31:0]) Host transmit RAM read data. This is the lower 4 bytes of transmit FIFO RAM data that is read at the address of hsttramradx[(TABITS-1):0] if hsttramradx[(TABITS+1)] is negated and hsttramrreq is asserted.This is a read-only field. Writes specifically to this field have no effects.

5.2.89. A-MCXFIFRAMAccess2 (Ask a Question)

Name: A-MCXFIFRAMAccess2
Offset: 0x068
Reset: 0x0
Property: Read/Write

A-MCXFIF FIFO RAM Access* Register 2

Bit	31	30	29	28	27	26	25	24
	HOST TRANSMIT RAM READ REQUEST (HSTTRAMRREQ)	HOST TRANSMIT RAM READ ACKNOWLEDGE (HSTTRAMRACK)						
Access	R/W	R						
Reset	0	0						

Bit	23	22	21	20	19	18	17	16
	HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[39:32])							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
				HOST TRANSMIT RAM READ ADDRESS (HSTTRAMRADX[12:0])				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST TRANSMIT RAM READ ADDRESS (HSTTRAMRADX[12:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 31 – HOST TRANSMIT RAM READ REQUEST (HSTTRAMRREQ) Host transmit RAM read request. Requests the handshake of hsttramradx values to the transmit FIFO RAM and hsttramrdat from the transmit FIFO RAM. Should only be asserted while hsttramrack is negated and while the transmit data path is disabled from receiving data and is in a steady state. It should only be negated after receiving an asserted hsttramrack.

Bit 30 – HOST TRANSMIT RAM READ ACKNOWLEDGE (HSTTRAMRACK) Host transmit RAM read acknowledge. Signifies the acceptance of hsttramradx values to the transmit FIFO RAM and reception of hsttramrdat from the transmit FIFO RAM location addressed. Will only be asserted or negated following assertion or negation of hsttramrreq. This is a read-only bit. Writes specifically to this bit have no effects.

Bits 23:16 – HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[39:32]) Host transmit RAM read data. This is the upper byte of transmit FIFO RAM data that was read at the address of hsttramwadx[(TABITS-1):0] if hsttramwadx[(TABITS+1)] is negated and hsttramwreq is asserted. This part of the transmit FIFO RAM contains control information for the frame as follows:

Value	Description
hsttramrdat[39]	1/0 - Control Frame/ non control frame.
hsttramrdat[38:37]	FIFO receive Per-Packet PAD Mode.
hsttramrdat[38:37]	FIFO receive Per-Packet PAD Mode.

HOST TRANSMIT RAM READ DATA (HSTTRAMRDAT[39:32]) (continued)	
Value	**Description**
hsttramrdat[35]	FIFO receive Per-Packet Generate FCS.
hsttramrdat[34]	FIFO receive end of frame.
hsttramrdat[33:32]	Valid byte enables and applicable only for the last word of the frame.
hsttramrdat[33:32] = 0	Indicates all bytes in the word are valid.
hsttramrdat[33:32] = 1	Indicates the LSB 3 bytes are valid [23:0] bits.
hsttramrdat[33:32] = 2	Indicates the LSB 2 bytes are valid [15:0] bits.
hsttramrdat[33:32] = 3	Indicates the LSB 1 bytes are valid [7:0] bits.

Bits 12:0 – HOST TRANSMIT RAM READ ADDRESS (HSTTRAMRADX[12:0]) Host transmit RAM read address. If hsttramradx[TABITS+1] is written low, hsttramradx[(TABITS-1):0] is the transmit FIFO RAM address which hsttramrdat is read from. If hsttramradx[TABITS+1] is written high, hsttramradx[TABITS:0] contains the pointer value read from system transmit module. The bit field length shown corresponds to a transmit RAM configured with 11 address bits (8 KB). The actual bit field width varies based on the configured transmit RAM size. The upper limit of this bit-field is derived from the TABITS parameter with the effective range as [(TABITS + 1) : 0].

5.2.90. A-MCXFIFRAMAccess1 (Ask a Question)

Name: A-MCXFIFRAMAccess1
Offset: 0x064
Reset: 0x0
Property: Read/Write

A-MCXFIF FIFO RAM Access* Register 1

Bit	31	30	29	28	27	26	25	24
	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[31:0]) Host transmit RAM write data. This is the lower 4 bytes of transmit FIFO RAM data that is written at the address of hsttramwadx[(TABITS-1):0] if hsttramwadx[(TABITS+1)] is negated and hsttramwreq is asserted.

5.2.91. A-MCXFIF5 (Ask a Question)

Name: A-MCXFIF5
Offset: 0x05C
Reset: 0xbfff7
Property: Read/Write

A-MCXFIF Configuration Register 5

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
		HALF DUPLEX INDICATOR (CFGHDPLX)	SYSTEM RECEIVE FIFO FULL (SRFULL)	HOST CLEAR SYSTEM RECEIVE FIFO FULL (HSTSRFULLCLR)	ONE BYTE TRANSFER PER SYSTEM CLOCK (CFGBYTMODE)	HOST DROP FRAMES LESS THAN 64 BYTES (HSTDRPLT64)	HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [17:16]	
Access		R/W	R	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	1	0	1	1

Bit	15	14	13	12	11	10	9	8
	HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	0	1	1	1

Bit 22 – HALF DUPLEX INDICATOR (CFGHDPLX) Assertion of this bit configures the MAC-FIFO to enable half-duplex back pressure as a flow control mechanism. De-assertion of this bit configures the MAC-FIFO to enable pause frames as a flow control mechanism.

Bit 21 – SYSTEM RECEIVE FIFO FULL (SRFULL) Assertion of this read-only bit indicates that the maximum capacity of the receive FIFO storage has been met or exceeded.

Bit 20 – HOST CLEAR SYSTEM RECEIVE FIFO FULL (HSTSRFULLCLR) This bit should be written asserted when it is desired to clear the srfull indicator bit. After hstfullclr assertion, srfull should be read until it becomes unasserted. Hstfullclr should then be written unasserted for the indicator to become operational again.

Bit 19 – ONE BYTE TRANSFER PER SYSTEM CLOCK (CFGBYTMODE) This bit should be asserted when data is transferred at the tpd and rpd bus at a rate of one byte per qualified clock. This bit should be negated when data is transferred at the tpd and rpd bus at a rate of one nibble per qualified clock. This bit should therefore be asserted when the MAC is configured for GMII mode.

Bit 18 – HOST DROP FRAMES LESS THAN 64 BYTES (HSTDRPLT64) Setting this bit causes the frame to be dropped if a receive frame is less than 64 bytes in length.

Bits 17:0 – HOST DONT CARE FOR FILTERING OF FRMES (HSTFLTRFRMDC) [17:0] The hstfltrfrmdc [17:0] configuration bits indicate which Receive Statistics Vectors are dont cares for MAC-FIFO frame drop circuitry. Setting of an hstfltrfrmdc bit, indicates a dont care for that hstfltrfrm bits. Clearing the bit looks for a matching level on the corresponding hstfltrfrm bit. If a match is made then the frame is dropped.

5.2.92. A-MCXFIFRAMAccess0 (Ask a Question)

Name: A-MCXFIFRAMAccess0
Offset: 0x060
Reset: 0x0
Property: Read/Write

A-MCXFIF FIFO RAM Access* Register 0

Bit	31	30	29	28	27	26	25	24
	HOST TRANSMIT RAM WRITE REQUEST (HSTTRAMWREQ)	HOST TRANSMIT RAM WRITE ACKNOWLEDGE (HSTTRAMWACK)						
Access	R/W	R						
Reset	0	0						

Bit	23	22	21	20	19	18	17	16
	HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[39:32])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
				HOST TRANSMIT RAM WRITE ADDRESS (HSTTRAMWADX [12:0])				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST TRANSMIT RAM WRITE ADDRESS (HSTTRAMWADX [12:0])							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 31 – HOST TRANSMIT RAM WRITE REQUEST (HSTTRAMWREQ) Host transmit RAM write request. Requests the handshake of hsttramwdat and hsttramwadx values to the transmit FIFO RAM. Should only be asserted while hsttramwack is negated and while the transmit data path is disabled from receiving data and is in a steady state. It should only be negated after receiving an asserted hsttramwack.

Bit 30 – HOST TRANSMIT RAM WRITE ACKNOWLEDGE (HSTTRAMWACK) Host transmit RAM write acknowledge. Signifies the acceptance of hsttramwdat and hsttramwadx values to the transmit FIFO RAM or FIFO Transmit module. Will only be asserted or negated following assertion or negation of hsttramwreq. This is a read-only bit. Writes specifically to this bit have no effects.

Bits 23:16 – HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[39:32]) Host transmit RAM write data. This is the upper byte of transmit FIFO RAM data that is written at the address of hsttramwadx [(TABITS-1):0] if hsttramwadx [TABITS+1] is negated and hsttramwreq is asserted. This part of the transmit FIFO RAM contains control information for the frame as follows:

Value	Description
hsttramwdat[39]	FIFO Transmit Control Frame (1b1 for control frame)
hsttramwdat[38]	Reserved
hsttramwdat[37]	FIFO Transmit Per-Packet PAD Mode
hsttramwdat[36]	FIFO Transmit Per-Packet enable

HOST TRANSMIT RAM WRITE DATA (HSTTRAMWDAT[39:32]) (continued)	
Value	**Description**
hsttramwdat[35]	FIFO Transmit Per-Packet Generate FCS
hsttramwdat[34]	FIFO Transmit end of packet
hsttramwdat[33:32]	FIFO Transmit data valid, applicable only for the last word of the frame
hsttramwdat[33:32] = 0	Indicates all bytes in the word are valid.
hsttramwdat[33:32] = 1	Indicates the LSB 3 bytes are valid [23:0] bits.
hsttramwdat[33:32] = 2	Indicates the LSB 2 bytes are valid [15:0] bits.
hsttramwdat[33:32] = 3	Indicates the LSB 1 bytes are valid [7:0] bits.

Bits 12:0 – HOST TRANSMIT RAM WRITE ADDRESS (HSTTRAMWADX [12:0]) Host transmit RAM write address. This field has different functionality based on the value of hsttramwadx [(TABITS+1)] and whether it is being written to or read from. When read from, hsttramwadx [TABITS:0] field contains the actual write pointer value of the FIFO Transmit module. When written to the hsttramwadx register is loaded. If hsttramwadx[TABITS+1] is low, hsttramwadx [(TABITS-1):0] transmits the RAM address which hsttramwdat is written to. If hsttramwadx [TABITS+1] is high, hsttramwadx [TABITS:0] contains the pointer value that is written to FIFO Transmit module. The bit field length shown corresponds to a transmit RAM configured with 11 address bits (8 KB). The actual bit field width varies based on the configured transmit RAM size. The upper limit of this bit-field is derived from the TABITS parameter with the effective range as [(TABITS + 1) : 0].

5.2.93. A-MCXFIF4 (Ask a Question)

Name: A-MCXFIF4
Offset: 0x058
Reset: 0x8
Property: Read/Write

A-MCXFIF Configuration Register 4

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							HOST FILTER FRAMES (HSTFLTRFRM) [17:16]	
Access							R/W	R/W
Reset							0	0

Bit	15	14	13	12	11	10	9	8
	HOST FILTER FRAMES (HSTFLTRFRM) [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	HOST FILTER FRAMES (HSTFLTRFRM) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	1	0	0	0

Bits 17:0 – HOST FILTER FRAMES (HSTFLTRFRM) [17:0]

Value	Description
17	Unicast frame detected but did not match configured station address.
16	Receive Frame Truncated.
15	Receive Long Event: CRS pin of TSE senses the activity over the channel. By enabling bit 15 of 0x058 long event packets received can be set to drop by TSE.
14	Receive VLAN Tag Detected: Frames length/type field contained 0x8100 which is the VLAN Protocol Identifier.
13	Receive Unsupported Op-code: Current Frame was recognized as a Control frame by the PEMCS, but it contains an unknown Op-code.
12	Receive PAUSE Control Frame: Current frame was recognized as a Control frame containing a valid PAUSE Frame Op-code and a valid address.
11	Receive Control Frame: Current Frame was recognized as a Control frame for having a valid Type-Length designation.
10	Receive Dribble Nibble: Indicates that after the end of the packet an additional 1 to 7 bits were received. A single nibble, called the dribble nibble, is formed but not sent to the system (10/100 Mbps only).
9	Receive Broadcast: Packets destination address contained the broadcast address.
8	Receive Multicast: Packets destination address contained a multicast address.

HOST FILTER FRAMES (HSTFLTRFRM) (continued)

Value	Description
7	Receive OK. Frame contained a valid CRC and did not have a code error.
6	Receive Length Out of Range: Indicates that frames length was larger than 1,518 bytes but smaller than the hosts maximum frame length value (type field).
5	Receive Length Check Error: Indicates that frame length field value in the packet does not match the actual data byte length and is not a type field.
4	Receive CRC Error: Packets CRC did not match the internally generated CRC.
3	Receive Code Error: One or more nibbles were signaled as errors during the reception of the packet.
2	Receive False Carrier: Indicates that at some time since the last receive statistics vector, a false carrier was detected, noted, and reported with this the next receive statistics. The false carrier is not associated with this packet. False carrier is an activity on the receive channel that does not result in a packet receive attempt being made. Defined to be RXER = 1, RXDV = 0, RXD[3:0] = 0xE (RXD[7:0] = 0x0E).
1	Receive RXDV Event: Indicates that the last receive event seen was not long enough to be a valid packet.
0	Receive Previous Packet Dropped as IFG is small.

5.2.94. A-MCXFIF3 (Ask a Question)

Name: A-MCXFIF3
Offset: 0x054
Reset: 0xfff0fff
Property: Read/Write

A-MCXFIF Configuration Register 3

Bit	31	30	29	28	27	26	25	24
					MAX NUMBER OF WORDS IN TRANSMIT FIFO (CFGHWMFT) [11:8]			
Access					R/W	R/W	R/W	R/W
Reset					1	1	1	1

Bit	23	22	21	20	19	18	17	16
	MAX NUMBER OF WORDS IN TRANSMIT FIFO (CFGHWMFT) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8
				FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH) [12:8]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bits 27:16 – MAX NUMBER OF WORDS IN TRANSMIT FIFO (CFGHWMFT) [11:0] After FIFO reaches the configured water mark level (cfghwmft), DMA engine stops reading data from system memory. Each hex value represents 4 byte locations that will be simultaneously stored in the transmit RAM. The bit field length shown corresponds to a transmit RAM configured with 11 address bits (8 KB). The actual bit field width varies based on the configured transmit RAM size. The upper limit of this bit-field is derived from the TABITS parameter with the effective range as [(TABITS + 16 − 1) : 16].

Bits 12:0 – FABRIC TRANSMIT CUT THROUGH THRESHOLD (CFGFTTH) [12:0] Once the transmit FIFO reaches the cut through threshold (cfgftth), MAC core informs to start frame transmission. Each hex value represents 4 byte locations that simultaneously stores in the transmit RAM. The bit field length shown corresponds to a transmit RAM configured with 11 address bits (8 KB). The actual bit field width varies based on the configured transmit RAM size. The upper limit of this bit-field is derived from the TABITS parameter with the effective range as [(TABITS + 1) : 0].

5.2.95. A-MCXFIF2 (Ask a Question)

Name: A-MCXFIF2
Offset: 0x050
Reset: 0x1fff1fff
Property: Read/Write

A-MCXFIF Configuration Register 2

Bit	31	30	29	28	27	26	25	24
				MAX WORDS IN RECEIVE FIFO (CFGHWM) [12:8]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				1	1	1	1	1

Bit	23	22	21	20	19	18	17	16
	MAX WORDS IN RECEIVE FIFO (CFGHWM) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8
				MIN WORDS IN RECEIVE FIFO BEFORE (CFGLWM) [12:8]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	MIN WORDS IN RECEIVE FIFO BEFORE (CFGLWM) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bits 28:16 – MAX WORDS IN RECEIVE FIFO (CFGHWM) [12:0] Once the receive FIFO reach the configured water mark level (cfghwm) MAC-FIFO sends XOFF pause control frame. Each hex value represents 4 byte locations that simultaneously stores in the receive RAM. The bit field length shown corresponds to a receive RAM configured with 12 address bits (8 KB). The actual bit field width varies based on the configured receive RAM size. The upper limit of this bit-field is derived from the RABITS parameter with the effective range as [(RABITS : 0].

Bits 12:0 – MIN WORDS IN RECEIVE FIFO BEFORE (CFGLWM) [12:0] Once the receive RAM reaches the cfglwm, XON (transmit ON) pause control frame transmits in response to a previously transmitted XOFF pause control frame. Each hex value represents 4 byte locations that simultaneously stores in the receive RAM. The bit field length shown corresponds to a receive RAM configured with 12 address bits (8 KB). The actual bit field width varies based on the configured receive RAM size. The upper limit of this bit-field is derived from the RABITS parameter with the effective range as [(RABITS : 0].

5.2.96. A-MCXFIF1 (Ask a Question)

Name: A-MCXFIF1
Offset: 0x04C
Reset: 0xffffffff
Property: Read/Write

A-MCXFIF Configuration Register 1

Bit	31	30	29	28	27	26	25	24
					SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH) [11:8]			
Access					R/W	R/W	R/W	R/W
Reset					1	1	1	1

Bit	23	22	21	20	19	18	17	16
	SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8
	NUMBER OF PAUSE QUANTAS BEFORE XOFF RETRANSMISSION (CFGXOFFRTX) [15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	NUMBER OF PAUSE QUANTAS BEFORE XOFF RETRANSMISSION (CFGXOFFRTX) [7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bits 27:16 – SYSTEM RECEIVE FOR CUT THROUGH THRESHOLD (CFGSRTH) [11:0] MAC-FIFO sub module will trigger DMA engine to start the DMA transfer(frame ready, frrdy) of received frame after reaching the threshold set in this bitfield. When set to maximum value, frrdy may be asserted only after the complete reception of the frame. The value of this register must be greater than 18d when hstdrplt64 is asserted. Each hex value represents 4 byte locations that will be simultaneously stored in the receive RAM. The bit field length shown corresponds to a receive RAM configured with 12 address bits (8 KB). The actual bit field width varies based on the configured receive RAM size. The upper limit of this bit-field is derived from the RABITS parameter with the effective range as [(RABITS -1 : 0].

Bits 15:0 – NUMBER OF PAUSE QUANTAS BEFORE XOFF RETRANSMISSION (CFGXOFFRTX) [15:0] This hex value represents the number of pause quanta (64 bit times) after an XOFF pause frame has been acknowledged until the MAC-FIFO re-asserts Transmit-Control Frame-Request (TCRQ) if the MAC-FIFO receive storage level has remained higher than the low watermark.

5.2.97. A-MCXFIF0 (Ask a Question)

Name: A-MCXFIF0
Offset: 0x048
Reset: 0x1f
Property: Read/Write

A-MCXFIF Configuration Register 0

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
				FABRIC TRANSMIT MODULE ENABLE STATUS (FTFENRPLY)	SYSTEM TRANSMIT MODULE ENABLE STATUS (STFENRPLY)	FABRIC RECEIVE MODULE ENABLE STATUS (FRFENRPLY)	SYSTEM RECEIVE MODULE ENABLE STATUS (SRFENRPLY)	WATER MARK MODULE ENABLE STATUS (WTMENRPLY)
Access				R	R	R	R	R
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
				FABRIC TRANSMIT MODULE ENABLE REQUEST (FTFENREQ)	SYSTEM TRANSMIT MODULE ENABLE REQUEST (STFENREQ)	FABRIC RECEIVE MODULE ENABLE REQUEST (FRFENREQ)	SYSTEM RECEIVE MODULE ENABLE REQUEST(SRFENREQ)	WATER MARK MODULE ENABLE REQUEST(WTMENREQ)
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
				HOST FABRIC TRANSMIT MODULE RESET (HSTRSTFT)	HOST MAC TRANSMIT MODULE RESET (HSTRSTST)	HOST FABRIC RECEIVE MODULE RESET (HSTRSTFR)	HOST MAC RECEIVE MODULE RESET (HSTRSTSR)	HOST TRANSMIT WATERMARK MODULE RESET (HSTRSTWT)
Access				R/W	R/W	R/W	R/W	R/W
Reset				1	1	1	1	1

Bit 20 – FABRIC TRANSMIT MODULE ENABLE STATUS (FTFENRPLY) When asserted, the Fabric transmit module is enabled. When negated, the Fabric transmit module is disabled. The bit should be polled until it reaches the expected value.

Bit 19 – SYSTEM TRANSMIT MODULE ENABLE STATUS (STFENRPLY) When asserted, the System transmit module is enabled. When negated, the System transmit module is disabled. The bit should be polled until it reaches the expected value.

Bit 18 – FABRIC RECEIVE MODULE ENABLE STATUS (FRFENRPLY) When asserted, the Fabric receive module is enabled. When negated, the Fabric receive module is disabled. The bit should be polled until it reaches the expected value.

Bit 17 – SYSTEM RECEIVE MODULE ENABLE STATUS (SRFENRPLY) When asserted, the System receive module is enabled and start of packet has been received. When negated, the System receive module is disabled and end-of-frame signal is received.

Bit 16 – WATER MARK MODULE ENABLE STATUS (WTMENRPLY) When asserted, the Water mark module is enabled. When negated, the Water mark module is disabled. The bit should be polled until it reaches the expected value.

Bit 12 – FABRIC TRANSMIT MODULE ENABLE REQUEST (FTFENREQ) When asserted, requests enabling of the Fabric transmit module. When negated, requests disabling of the Fabric transmit module.

Bit 11 – SYSTEM TRANSMIT MODULE ENABLE REQUEST (STFENREQ) When asserted, requests enabling of the System transmit module.

Bit 10 – FABRIC RECEIVE MODULE ENABLE REQUEST (FRFENREQ) When asserted, requests enabling of the Fabric receive module.

Bit 9 – SYSTEM RECEIVE MODULE ENABLE REQUEST(SRFENREQ) When asserted, requests enabling of the System receive module. When negated, requests disabling of the System receive module.

Bit 8 – WATER MARK MODULE ENABLE REQUEST(WTMENREQ) When asserted, requests enabling of the Water mark module. When negated, requests disabling of tark module.

Bit 4 – HOST FABRIC TRANSMIT MODULE RESET (HSTRSTFT) When asserted this bit places fabric transmit module in reset.

Bit 3 – HOST MAC TRANSMIT MODULE RESET (HSTRSTST) When asserted this bit places the System transmit module in reset.

Bit 2 – HOST FABRIC RECEIVE MODULE RESET (HSTRSTFR) When asserted this bit places the fabric receive module in reset.

Bit 1 – HOST MAC RECEIVE MODULE RESET (HSTRSTSR) When asserted this bit places the System receive module in reset.

Bit 0 – HOST TRANSMIT WATERMARK MODULE RESET (HSTRSTWT) When asserted this bit places the transmit watermark module in reset.

5.3. TBI/1000Base-X Registers (Indirect Addressing through MDIO) (Ask a Question)

Configuration and status of the core is achieved by the management registers accessed through the serial MDIO. The TBI core is a dual mode core and to operate in TBI or 1000Base-X mode is user selectable.

The following registers are common in both modes:

- Control register at address 0x00
- Status register at address 0x01
- Extended Status register at address 0x0F
- Jitter Diagnostic register at address 0x10
- TBI Control register at address 0x11

Other registers (at address 0x04, 0x05, 0x06, 0x07, and 0x08) are based on mode selected. In 0, these registers are described separately.

All these 16-bit registers are associated with MDIO functionality and are accessed indirectly through the serial MDIO interface. To access these MDIO registers using the AHB interface, perform the following steps:

1. Writing to MDIO Registers:
   Configure the "MDIO Mgmt: Address (0x028)" register and the "MDIO Mgmt: Control (0x02C)" register with the appropriate values.
2. Reading from MDIO Registers:
   Configure the "MDIO Mgmt: Address (0x028)" register and the "MDIO Mgmt: Command (0x024)" register with the appropriate values. The data from the selected MDIO register will be available in the "MDIO Mgmt: Status (0x030)" register, which should then be read.

5.4. MDIO Register Summary (Ask a Question)

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	Control	7:0								
		15:8	PHY RESET	LOOP BACK		AUTO-NEGOTIATION ENABLE			RESTART AUTO-NEGOTIATION	
0x01	Status	7:0		MF PREMABLE SUPPRESSION ENABLE	AUTO-NEGOTIATION COMPLETE	REMOTE FAULT	AUTO-NEGOTIATION ABILITY	LINK STATUS		EXTENDED CAPABILITY
		15:8								EXTENDED STATUS
0x03	Reserved									
0x04	AN_Advertisement (1000BASE-X)	7:0	PAUSE[0]	HALF-DUPLEX	FULL-DUPLEX					
		15:8	NEXT PAGE		REMOTE FAULT[1:0]					PAUSE[1]
0x04	AN_Advertisement (1000BASE-T)	7:0	TBI OPERATION[7:0]							
		15:8	LINK UP	ACK		FULL-DUPLEX	LINK SPEED[1:0]		TBI OPERATION[9:8]	
0x05	AN_Link_Partner_Base_Page_Ability (1000BASE-X)	7:0	PAUSE[0]	HALF-DUPLEX	FULL-DUPLEX					
		15:8	NEXT PAGE		REMOTE FAULT[1:0]					PAUSE[1]
0x05	AN_Link_Partner_Base_Page_Ability (1000BASE-T)	7:0	TBI OPERATION[7:0]							
		15:8	LINK UP	AUTO-NEGOTIATION ACK AS SPECIFIED IN 802.3Z		FULL-DUPLEX	LINK SPEED[1:0]		TBI OPERATION[9:8]	
0x06	AN_Expansion (1000BASE-X)	7:0						NEXT PAGE ABLE	PAGE RECEIVED	
		15:8								
0x06	AN_Expansion (1000BASE-T)	7:0						NEXT PAGE ABLE	PAGE RECEIVED	
		15:8								
0x07	AN_Next_Page_Transmit (1000BASE-X)	7:0	MESSAGE/UNFORMATTED CODE FIELD[7:0]							
		15:8	NEXT PAGE		MESSAGE PAGE	ACKNOWLEDGE 2	TOGGLE	MESSAGE/UNFORMATTED CODE FIELD[10:8]		
0x07	AN_Next_Page_Transmit (1000BASE-T)	7:0	USER DEFINED REGISTER[7:0]							
		15:8	USER DEFINED REGISTER[15:8]							
0x08	AN_Link_Partner_Ability_Next_Page (1000BASE-X)	7:0	MESSAGE OR UNFORMATTED CODE FIELD[7:0]							
		15:8	NEXT PAGE		MESSAGE PAGE	ACKNOWLEDGE 2	TOGGLE	MESSAGE OR UNFORMATTED CODE FIELD[10:8]		
0x08	AN_Link_Partner_Ability_Next_Page (1000BASE-T)	7:0	USER DEFINED REGISTER[7:0]							
		15:8	USER DEFINED REGISTER[15:8]							
0x0A ... 0x0E	Reserved									
0x0F	Extended_Status	7:0								
		15:8	1000BASE-X FULL-DUPLEX	1000BASE-X HALF-DUPLEX	1000BASE-T FULL-DUPLEX	1000BASE-T HALF-DUPLEX				
0x10	Jitter_Diagnostics	7:0	CUSTOM JITTER PATTERN[7:0]							
		15:8	JITTER DIAGNOSTIC ENABLE	JITTER PATTERN SELECT[2:0]					CUSTOM JITTER PATTERN[9:8]	
0x11	SGMII Ten_Bit_Interface_Control	7:0								
		15:8	SOFT RESET	SHORTCUT LINK TIMER	DISABLE RECEIVE RUNNING DISPARITY	DISABLE TRANSMIT RUNNING DISPARITY	GO LINK TIMER VALUE CONTROL			AUTO-NEGOTIATION SENSE

5.4.1. Status (Ask a Question)

Name: Status
Offset: 0x001
Reset: 0x149
Property: Read-only

Status

Bit	15	14	13	12	11	10	9	8
								EXTENDED STATUS
Access								R
Reset								1

Bit	7	6	5	4	3	2	1	0
		MF PREMABLE SUPPRESSION ENABLE	AUTO-NEGOTIATION COMPLETE	REMOTE FAULT	AUTO-NEGOTIATION ABILITY	LINK STATUS		EXTENDED CAPABILITY
Access		R	R	R	R	R		R
Reset		1	0	0	1	0		1

Bit 8 – EXTENDED STATUS This bit indicates that PHY status information is also contained in Register 0x0F:EXTENDED STATUS.

Bit 6 – MF PREMABLE SUPPRESSION ENABLE This bit indicates whether the PHY is capable of handling MDIO management Frames without the 32-bit preamble field. Returns 1 indicating support for suppressed preamble MDIO management Frames.

Bit 5 – AUTO-NEGOTIATION COMPLETE When 1, this bit indicates that the auto-negotiation process has completed. This bit returns 0 when either the auto-negotiation process is underway or when the auto-negotiation function is disabled.

Bit 4 – REMOTE FAULT When 1, a remote fault condition has been detected between the TBI and the PHY.

Bit 3 – AUTO-NEGOTIATION ABILITY When 1, this bit indicates that the TBI has the ability to perform auto-negotiation.

Bit 2 – LINK STATUS When 1, this bit indicates that a valid link has been established between the TBI and the PHY. When 0, no valid link has been established.

Bit 0 – EXTENDED CAPABILITY This bit indicates that the TBI contains the extended set of registers.

5.4.2. SGMII Ten_Bit_Interface_Control (Ask a Question)

Name: SGMII Ten_Bit_Interface_Control
Offset: 0x011
Reset: 0x0
Property: Read/Write

SGMII/Ten Bit Interface Control

Bit	15	14	13	12	11	10	9	8
	SOFT RESET	SHORTCUT LINK TIMER	DISABLE RECEIVE RUNNING DISPARITY	DISABLE TRANSMIT RUNNING DISPARITY	GO LINK TIMER VALUE CONTROL			AUTO-NEGOTIATION SENSE
Access	R/W	R/W	R/W	R/W	R/W			R/W
Reset	0	0	0	0	0			0

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

Bit 15 – SOFT RESET This bit resets the functional modules in the TBI. Clear it for normal operation.

Bit 14 – SHORTCUT LINK TIMER Set this bit 1 to reduce the value of Go Link Timer and Sync. Status Fail Timer to 64 clock pulse. Ultimately this reduces the amount of simulation time needed to time the 1.6 ms Link Timer. Clear it for normal operation. In normal operation the value of Go Link Timer is 200000 clock pulses and the value of the Sync. Status Fail Timer is 1250000 clock pulses.

Bit 13 – DISABLE RECEIVE RUNNING DISPARITY Set this bit to disable the running disparity calculation and checking in the receive direction. This bit must be 0 for TBI operation

Bit 12 – DISABLE TRANSMIT RUNNING DISPARITY Set this bit to disable the running disparity calculation and checking in the transmit direction. This bit must be 0 for TBI operation.

Bit 11 – GO LINK TIMER VALUE CONTROL When 0, the Go Link Timer Value = 1.6 ms. When set to 1, the Go Link Timer Value = 10 ms.

Bit 8 – AUTO-NEGOTIATION SENSE Set this bit to allow the auto-negotiation function to sense either a MAC in Auto-Negotiation bypass mode or an older MAC without auto-negotiation capability. When sensed, Auto- Negotiation Complete becomes true; however Page Received is low, indicating no page was exchanged. Management can then act accordingly. Clear this bit when IEEE 802.3z Clause 37 behaviour is desired, which results in the link not coming up.

5.4.3. Jitter_Diagnostics (Ask a Question)

Name: Jitter_Diagnostics
Offset: 0x010
Reset: 0x0
Property: Read/Write

Jitter Diagnostics

Bit	15	14	13	12	11	10	9	8
	JITTER DIAGNOSTIC ENABLE	JITTER PATTERN SELECT[2:0]					CUSTOM JITTER PATTERN[9:8]	
Access	R/W	R/W	R/W	R/W			R/W	R/W
Reset	0	0	0	0			0	0

Bit	7	6	5	4	3	2	1	0
	CUSTOM JITTER PATTERN[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 15 – JITTER DIAGNOSTIC ENABLE Set this bit to enable the TBI to transmit the jitter test patterns defined in IEEE 802.3z 36A. Clear this bit to enable normal transmit-operation.

Bits 14:12 – JITTER PATTERN SELECT[2:0]

Value	Description		
Jitter Pattern Select	Bit 14	Bit 13	Bit 12
User Defined Custom Pattern	0	0	0
Annex 36A Defined High Frequency 10101010101010101010€¦	0	0	1
Annex 36A Defined Mixed Frequency 11111010110000010100€¦	0	1	0
Custom Defined Low Frequency 11111000001111100000€¦	0	1	1
Random Jitter Pattern	1	0	0
Annex 36A Defined Low Frequency 11111000001111100000€¦ 11111000001111100000€¦	1	0	1
Reserved	1	1	0
Reserved	1	1	1

Bits 9:0 – CUSTOM JITTER PATTERN[9:0] Used in conjunction with JITTER PATTERN SELECT and JITTER DIAGNOSTIC ENABLE. Set this field to the desired custom pattern, which transmits continuously.

5.4.4. Extended_Status (Ask a Question)

Name: Extended_Status
Offset: 0x00F
Reset: 0xa000
Property: Read-only

Extended Status

Bit	15	14	13	12	11	10	9	8
	1000BASE-X FULL-DUPLEX	1000BASE-X HALF-DUPLEX	1000BASE-T FULL-DUPLEX	1000BASE-T HALF-DUPLEX				
Access	R	R	R	R				
Reset	1	0	1	0				

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

Bit 15 – 1000BASE-X FULL-DUPLEX When 1, indicates PHY can operate in 1000BASE-X Full-Duplex mode. When 0, indicates PHY cannot operate in this mode.

Bit 14 – 1000BASE-X HALF-DUPLEX When 1, indicates PHY can operate in 1000BASE-X Half-Duplex mode. When 0, indicates PHY cannot operate in this mode.

Bit 13 – 1000BASE-T FULL-DUPLEX

Bit 12 – 1000BASE-T HALF-DUPLEX When 1, indicates PHY can operate in 1000BASE-T Half-Duplex mode. When 0, indicates PHY cannot operate in this mode.

5.4.5. Control (Ask a Question)

Name: Control
Offset: 0x000
Reset: 0x0
Property: Read/Write

Control

Bit	15	14	13	12	11	10	9	8
	PHY RESET	LOOP BACK		AUTO-NEGOTIATION ENABLE			RESTART AUTO-NEGOTIATION	
Access	R/W	R/W		R/W			R/W	
Reset	0	0		0			0	

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

Bit 15 – PHY RESET Setting this bit causes the Tx, Rx, and AutoNegX sub-modules in the TBI core to be reset. This bit is self-clearing.

Bit 14 – LOOP BACK Setting this bit causes the 10-bit transmit outputs of the TBI to be connected to the receive 10-bit inputs. Clearing this bit results in normal operation. This bit does not affect the clock signals, which for loopback must be handled external to the core.

Bit 12 – AUTO-NEGOTIATION ENABLE Setting this bit enables the auto-negotiation process. If cleared, then the values programmed determines the operating condition of the link.

Bit 9 – RESTART AUTO-NEGOTIATION Setting this bit causes the auto-negotiation process to restart. This action is only available when auto-negotiation has been enabled.

5.4.6. AN_Next_Page_Transmit (1000BASE-X) (Ask a Question)

Name: AN_Next_Page_Transmit (1000BASE-X)
Offset: 0x007
Reset: 0x0
Property: Read/Write

AN Next Page Transmit (1000BASE-X)

Bit	15	14	13	12	11	10	9	8
	NEXT PAGE		MESSAGE PAGE	ACKNOWLEDGE 2	TOGGLE	MESSAGE/UNFORMATTED CODE FIELD[10:8]		
Access	R/W		R/W	R/W	R	R	R	R
Reset	0		0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	MESSAGE/UNFORMATTED CODE FIELD[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit 15 – NEXT PAGE Assert this bit to indicate additional Next Pages to follow. Bit is cleared to indicate last page.

Bit 13 – MESSAGE PAGE Assert bit to indicate Message Page. Clear bit to indicate Unformatted Page.

Bit 12 – ACKNOWLEDGE 2 Used by Next Page function to indicate device has ability to comply with the message. Assert bit if local device complies with message. Clear bit if local device cannot comply with message.

Bit 11 – TOGGLE Used to ensure synchronization with the Link Partner during Next Page exchange. This bit always takes opposite value of the Toggle bit of the previously exchanged Link Code Word. The initial value in the first Next Page transmitted is the inverse of bit 11 in the base Link Code Word.

Bits 10:0 – MESSAGE/UNFORMATTED CODE FIELD[10:0] Message pages are formatted pages that carry a predefined Message Code, which is enumerated in IEEE 802.3u/Annex 28C. Unformatted Code Fields take on an arbitrary value.

5.4.7. AN_Next_Page_Transmit (1000BASE-T) (Ask a Question)

Name: AN_Next_Page_Transmit (1000BASE-T)
Offset: 0x007
Reset: 0x0
Property: Read/Write

Use of this register is user dependent. User can define functionality of bits of this register as per system requirement.

Bit	15	14	13	12	11	10	9	8
	USER DEFINED REGISTER[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	USER DEFINED REGISTER[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – USER DEFINED REGISTER[15:0]

5.4.8. AN_Link_Partner_Base_Page_Ability (1000BASE-X) (Ask a Question)

Name: AN_Link_Partner_Base_Page_Ability (1000BASE-X)
Offset: 0x005
Reset: 0x0
Property: Read-only

AN Link Partner Base Page Ability (1000BASE-X)

Bit	15	14	13	12	11	10	9	8
	NEXT PAGE		REMOTE FAULT[1:0]					PAUSE[1]
Access	R		R	R				R
Reset	0		0	0				0

Bit	7	6	5	4	3	2	1	0
	PAUSE[0]	HALF-DUPLEX	FULL-DUPLEX					
Access	R	R	R					
Reset	0	0	0					

Bit 15 – NEXT PAGE The Link Partner asserts this bit either to request Next Page transmission or to indicate the capability to receive Next Pages. When 0, the Link Partner has no subsequent Next Pages or is not capable of receiving Next Pages.

Bits 13:12 – REMOTE FAULT[1:0] Encodes the Link Partners remote fault condition.

Value	Description	
RF1(4.12)	RF2(4.13)	Description
0	0	No error, link Ok.
0	1	Offline.
1	0	Link_Failure.
1	1	Auto-Negotiation_Error.

Bits 8:7 – PAUSE[1:0]

Value	Description	
PAUSE1(4.7)	ASM_DIR(4.8)	Capability
0	0	No PAUSE.
0	1	Asymmetric PAUSE toward link partner.
1	0	Symmetric PAUSE.
1	1	Both Symmetric PAUSE and Asymmetric PAUSE toward local device.

Bit 6 – HALF-DUPLEX When 1, Link Partner is capable of half-duplex operation. When 0, Link Partner is incapable of half-duplex mode.

Bit 5 – FULL-DUPLEX When 1, Link Partner is capable of full-duplex operation. When 0, Link Partner is incapable of full-duplex mode.

5.4.9. AN_Link_Partner_Base_Page_Ability (1000BASE-T) (Ask a Question)

Name: AN_Link_Partner_Base_Page_Ability (1000BASE-T)
Offset: 0x005
Reset: 0x0
Property: Read/Write

AN Link Partner Base Page Ability (1000BASE-T)

Bit	15	14	13	12	11	10	9	8
	LINK UP	AUTO-NEGOTIATION ACK AS SPECIFIED IN 802.3Z		FULL-DUPLEX	LINK SPEED[1:0]		TBI OPERATION[9:8]	
Access	R	R		R	R	R	R/W	R/W
Reset	0	0		0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TBI OPERATION[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 15 – LINK UP Assertion of this bit indicates that the link is up.

Bit 14 – AUTO-NEGOTIATION ACK AS SPECIFIED IN 802.3Z

Bit 12 – FULL-DUPLEX Assertion of this bit indicates that the link is transferring data in Full-Duplex mode.

Bits 11:10 – LINK SPEED[1:0] Assertion of these 2 bits indicates the speed that the link is transferring data.

Value	Description
2b00	10Mbps
2b01	100Mbps
2b10	1000Mbps
2b11	Reserve

Bits 9:0 – TBI OPERATION[9:0] These bits must always be written 0000000001 for TBI operation.

5.4.10. AN_Link_Partner_Ability_Next_Page (1000BASE-X) (Ask a Question)

Name: AN_Link_Partner_Ability_Next_Page (1000BASE-X)
Offset: 0x008
Reset: 0x0
Property: Read-only

AN Link Partner Ability Next Page (1000BASE-X)

Bit	15	14	13	12	11	10	9	8
	NEXT PAGE		MESSAGE PAGE	ACKNOWLEDGE 2	TOGGLE	MESSAGE OR UNFORMATTED CODE FIELD[10:8]		
Access	R		R	R	R	R	R	R
Reset	0		0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	MESSAGE OR UNFORMATTED CODE FIELD[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit 15 – NEXT PAGE The Link Partner asserts this bit to indicate additional Next Pages to follow. When 0, indicates last Next Page from link partner.

Bit 13 – MESSAGE PAGE When 1, indicates Message Page. When 0, indicates Unformatted Page.

Bit 12 – ACKNOWLEDGE 2 Indicates Link Partners ability to comply with the message. When 1, Link Partner complies with message. When 0, Link Partner cannot comply with message.

Bit 11 – TOGGLE Used to ensure synchronization with the Link Partner during Next Page exchange. This bit always takes opposite value of the Toggle bit of the previously exchanged Link Code Word. The initial value in the first Next Page transmitted is the inverse of bit 11 in the base Link Code Word.

Bits 10:0 – MESSAGE OR UNFORMATTED CODE FIELD[10:0] Message pages are formatted pages that carry a predefined Message Code, which is enumerated in IEEE 802.3u/Annex 28C. Unformatted Code Fields take on an arbitrary value.

5.4.11. AN_Link_Partner_Ability_Next_Page (1000BASE-T) (Ask a Question)

Name: AN_Link_Partner_Ability_Next_Page (1000BASE-T)
Offset: 0x008
Reset: 0x0
Property: Read-only

Use of this register is user dependent. User can define functionality of bits of this register as per system requirement.

Bit	15	14	13	12	11	10	9	8
	USER DEFINED REGISTER[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	USER DEFINED REGISTER[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – USER DEFINED REGISTER[15:0]

5.4.12. AN_Expansion (1000BASE-X) (Ask a Question)

Name: AN_Expansion (1000BASE-X)
Offset: 0x006
Reset: 0x4
Property: Read-only

AN Expansion (1000BASE-X)

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
						NEXT PAGE ABLE	PAGE RECEIVED	
Access						R	R	
Reset						1	0	

Bit 2 – NEXT PAGE ABLE When 1, indicates that the local device supports the Next Page function. Returns 1 on read.

Bit 1 – PAGE RECEIVED 1 indicates that a new page has been received and stored in the applicable AN LINK PARTNER ABILITY or AN NEXT PAGE register. This bit latches high in order for software to detect when polling. The bit is cleared on a read to the register.

5.4.13. AN_Advertisement (1000BASE-X) (Ask a Question)

Name: AN_Advertisement (1000BASE-X)
Offset: 0x004
Reset: 0x0
Property: Read/Write

AN_Advertisement (1000BASE-X)

Bit	15	14	13	12	11	10	9	8
	NEXT PAGE		REMOTE FAULT[1:0]					PAUSE[1]
Access	R/W		R/W	R/W				R/W
Reset	0		0	0				0

Bit	7	6	5	4	3	2	1	0
	PAUSE[0]	HALF-DUPLEX	FULL-DUPLEX					
Access	R/W	R/W	R					
Reset	0	0	0					

Bit 15 – NEXT PAGE The local device asserts this bit to either request Next Page transmission or advertise Next Page exchange capability. This bit can thus be set when the local has no Next Pages but wishes to allow reception of Next Pages. If the local device has no Next Pages, and the Link Partner wishes to send Next Pages, the local device should send Null Message Codes and have the MESSAGE PAGE set to 0b000_0000_0001. This bit should be cleared where the local device wishes not to engage in Next Page exchange.

Bits 13:12 – REMOTE FAULT[1:0]

Value	Description	
RF1(4.12)	RF2(4.13)	Description
0	0	No error, link Ok.
0	1	Offline.
1	0	Link_Failure.
1	1	Auto-Negotiation_Error.

Bits 8:7 – PAUSE[1:0] Encodes the local devices PAUSE capability.

Value	Description	
PAUSE1(4.7)	ASM_DIR(4.8)	Capability
0	0	No PAUSE.
0	1	Asymmetric PAUSE toward link partner.
1	0	Symmetric PAUSE.
1	1	Asymmetric PAUSE toward local device.

Bit 6 – HALF-DUPLEX Setting this bit means local device is capable of half-duplex operation.

Bit 5 – FULL-DUPLEX Setting this bit means local device is capable of full-duplex operation.

5.4.14. AN_Expansion (1000BASE-T) (Ask a Question)

Name: AN_Expansion (1000BASE-T)
Offset: 0x006
Reset: 0x4
Property: Read-only

AN Expansion (1000BASE-T)

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
						NEXT PAGE ABLE	PAGE RECEIVED	
Access						R	R	
Reset						1	0	

Bit 2 – NEXT PAGE ABLE When 1, indicates that the local device supports the Next Page function.

Bit 1 – PAGE RECEIVED When 1, indicates that a new page has been received and stored in the applicable AN LINK PARTNER ABILITY or AN NEXT PAGE.

5.4.15. AN_Advertisement (1000BASE-T) (Ask a Question)

Name: AN_Advertisement (1000BASE-T)
Offset: 0x004
Reset: 0x0
Property: Read/Write

AN Advertisement (1000BASE-T)

Bit	15	14	13	12	11	10	9	8
	LINK UP	ACK		FULL-DUPLEX	LINK SPEED[1:0]		TBI OPERATION[9:8]	
Access	R/W	R		R/W	R/W	R/W	R	R
Reset	0	0		0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TBI OPERATION[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit 15 – LINK UP This bit must be written 0 for TBI operation.

Bit 14 – ACK Reserved. Ignore on read.

Bit 12 – FULL-DUPLEX This bit must be written 0 for TBI operation.

Bits 11:10 – LINK SPEED[1:0] These bits must be written 00 for TBI operation.

Bits 9:0 – TBI OPERATION[9:0] These bits must always be written 0000000001 for TBI operation.

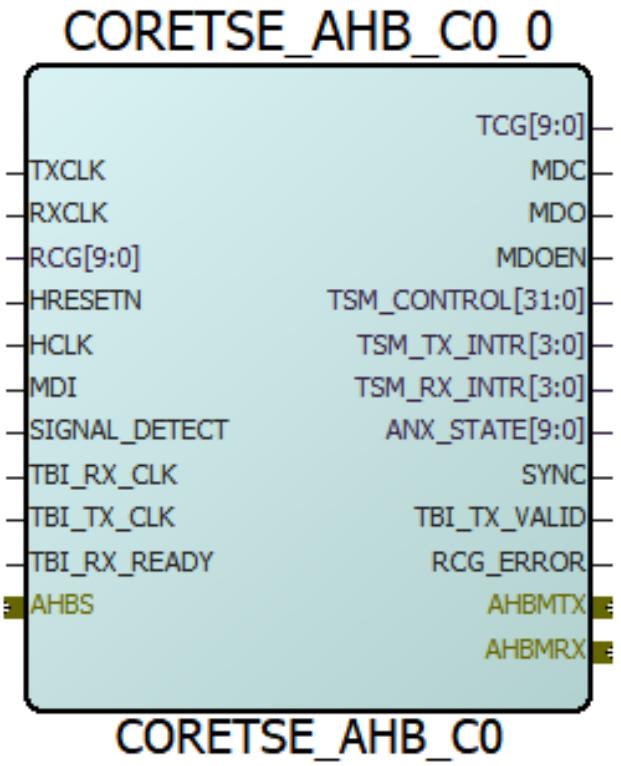
6. Implementing CoreTSE_AHB in Libero Design Suite (Ask a Question)

This section describes implementation of CoreTSE_AHB in Libero Design Suite.

6.1. SmartDesign (Ask a Question)

Once installed in the Libero software, the core can be instantiated, configured, connected and generated using the SmartDesign tool. An example instantiated view is shown in the following figure.

Figure 6-1. SmartDesign CoreTSE_AHB Instance View

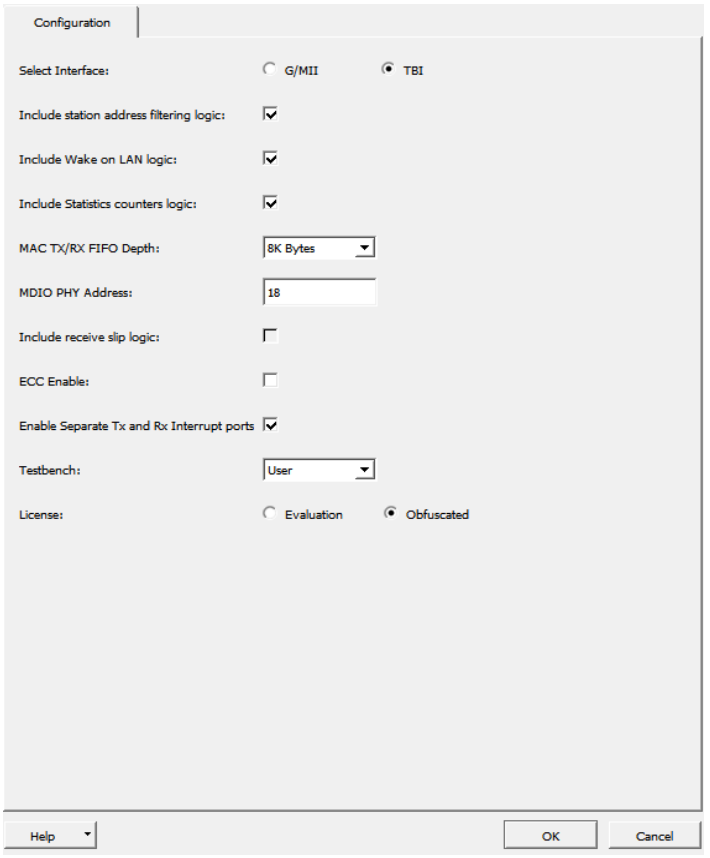


For more information on using SmartDesign to instantiate and generate cores, see the latest *DS50003075, SmartDesign User Guide* on Libero SoC Documentation page.

6.2. Configuring CoreTSE_AHB in SmartDesign (Ask a Question)

The CoreTSE_AHB configuration GUI takes up a large amount of screen area when it is sized to show all configuration options. An example of the GUI for the SmartFusion 2 family is shown in the following figure.

Figure 6-2. CoreTSE_AHB Configuration GUI



6.3. Simulation (Ask a Question)

To simulate the core using the testbench, after configuring the CoreTSE_AHB as per the requirement, perform the following steps:

1. Set the design root to the CoreTSE_AHB instantiation in the Libero® SoC **Design Hierarchy** pane, and then click **Build Hierarchy**.
2. Click the **Stimulus Hierarchy** tab, and then click **Build Hierarchy**.
3. Right-click testbench, under **Simulate Pre-Synthesis Design**, select **Open Interactively**.
4. For complete visibility of CoreTSE_AHB input and output ports, add an additional vsim command. To add command, perform the following steps:
   a) Navigate to **Project** > **Project Settings** > **Simulation Options** > **Vsim Commands**.
   b) Under Additional options, add "`-voptargs=+acc`".
   c) Click **Save Changes**

   QuestaSim opens with the testbench file and automatically runs the simulation.

Important: If the simulation is interrupted due to the runtime limit specified in the `.do` file, use the `run -all` command to complete the simulation.

6.4. Synthesis and Compile (Ask a Question)

To run synthesis, perform the following steps:

1. With the configuration selected in the configuration GUI, set the design root appropriately.
2. Under **Implement Design**, on the **Design Flow** tab, right-click **Synthesize** and click **Run**.

6.5. Place and Route (Ask a Question)

To run the place and route, perform the following steps:

1. With the configuration selected in the configuration GUI, set the design root appropriately.
2. Under **Implement Design**, on the **Design Flow** tab, right-click **Place and Route** and click **Run**.

7. Design Constraints (Ask a Question)

This section describes the design constraints of the CoreTSE_AHB.

7.1. Timing Constraints (Ask a Question)

This section provides the information on the timing constraints required for CoreTSE_AHB.

Important: In all the timing constraints provided in the reference sdc file, 'CORETSE_AHB_C0_0' is the instance name of CoreTSE_AHB IP in the Libero SoC . You can replace the instance name when using this timing constraint in the your design.

To meet the CoreTSE_AHB timing requirement, it is important to provide timing constraint.

MDC: Management clock output for MDIO interface. This clock is generated internally by the IP Core. HCLK is the source clock.

It is recommended to use the generated clock constraints for the management clock. HCLK is the source clock. The `MGMT CLOCK SELECT` field of the MDIO Mgmt: Configuration register (address 0x020) determines the frequency of this clock. An example is given in the reference `sdc` file for generated clock constraint for division factor 8 (MGMT CLOCK SELECT = 3'b011).

It is recommended to provide set clock group timing constraint to set false path between asynchronous clocks.

CoreTSE_AHB uses the following clocks when the **Select Interface** is set to **TBI**:

- HCLK
- TBI_TX_CLK
- TBI_RX_CLK
- TXCLK (Synchronous to TBI_TX_CLK)
- RXCLK (Synchronous to TBI_RX_CLK)
- MDC (Synchronous to HCLK)

When **Select Interface** is set to **GMII**:

- HCLK
- TXCLK
- RXCLK

The source clock name used in the reference sdc file are for reference only. The source name may be different in your design, and you must change the clock name in the constraint accordingly.

You can find reference sdc timing constraint file from the following path.

```
<project_directory/component/Actel/DirectCore/CoreTSE_AHB/(example
        4.0.xxx)/constraints/CoreTSE_AHB.sdc>
```

Important: Additionally, the generated clock (mdc) clock constraint can be found in the `CoreTSE_AHB_mdc.sdc` file, which is situated in the previously specified directory. It is important to ensure that the **Synthesis** option is unchecked in the Constraints Manager when utilizing the `CoreTSE_AHB_mdc.sdc` file.

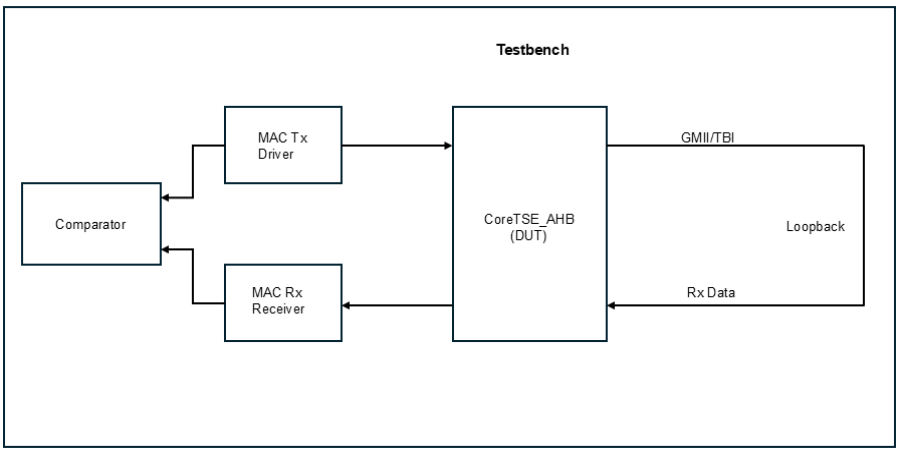
8. Testbench (Ask a Question)

A unified testbench is used to verify and test CoreTSE_AHB called as user test-bench.

8.1. User Testbench (Ask a Question)

A simplified block diagram of the user testbench is shown in the following figure.

Figure 8-1. User Testbench



The user testbench instantiates the CoreTSE_AHB as well as behavioral, non-synthesizable models of an input test generator and provides necessary clock, reset, and other signals. The testbench compares the actual CoreTSE_AHB output interface against the predicted behavioral model data.

The same testbench can be used for pre-synthesis and post-synthesis simulation. A simulation tool displays the verification result. Test bench has task based library models for AHB transmit, AHB receive, MAC link transmit, MAC link receive, and generic test bench to check and report errors.

Following are the modules instances used in testbench top:

- gl: Generic Library instantiations
- mlat: MAC library for MAC Data transmit
- mllr: MAC library for link receive
- mllt: MAC library for link transmit
- mlar: MAC library for MAC Data receive
- MAC library module has tasks related to MAC functionality
  - Tasks for generating descriptors
  - Tasks related to 8bto10b conversation and 10bto8b conversion logic
- Module gl has tasks related to generic test bench functionality
  - Tasks for simulation end, error message prints, and maximum error definitions.

In TBI mode, the following test case is available:

- Random and data, IPG, and preamble pattern test
  - This tests the DMA's random size and payload data frames through Transmit and receives the paths of the CoreTSE_AHB.

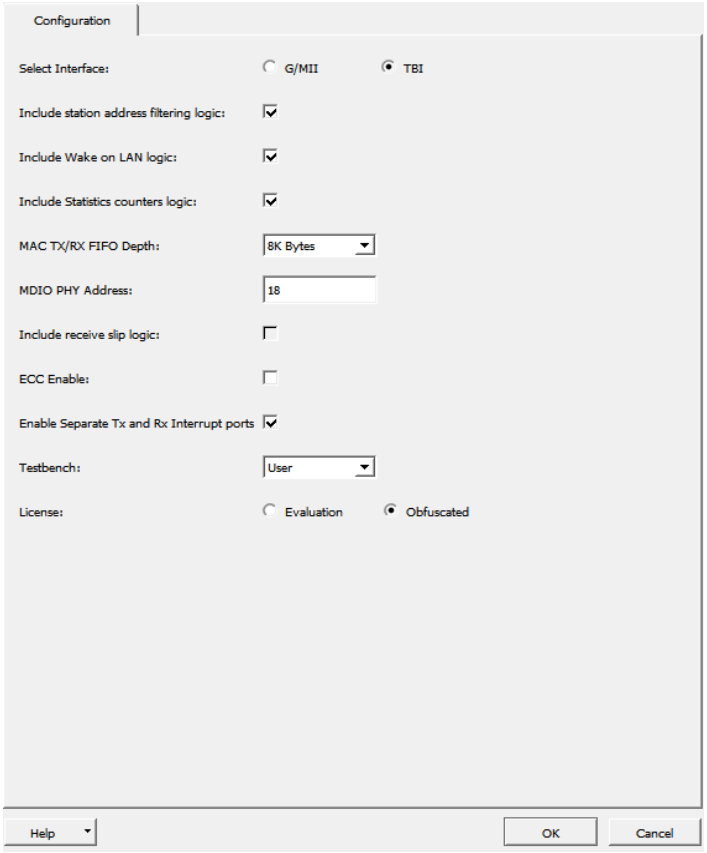
In G/MII mode, the following test cases are available:

- Random Data and Timing test

  - This tests the DMA's random size and payload data frames through transmit and then receive paths of the CoreTSE_AHB. This is done for different speed modes.
- Per Packet configuration test
  - This test enables Per Packet configuration of transmit path data and then verifies the operation of all bits at the link.

Simulation with "user testbench" passes only with default configuration, as shown in the following figure:

Figure 8-2. Default Configuration



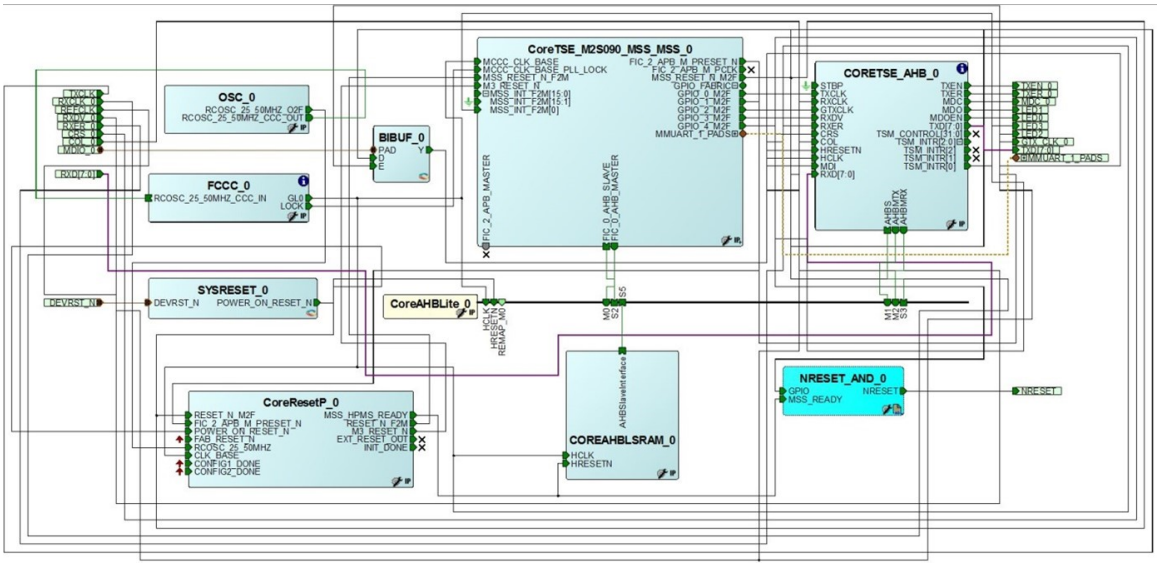
The **Select Interface** supports switching between G/MII and TBI.

9. System Integration (Ask a Question)

This section provides information to simplify the integration of CoreTSE_AHB.

The following figure shows the CoreTSE_AHB integration in G/MII system.

Figure 9-1. CoreTSE_AHB Integration in G/MII System



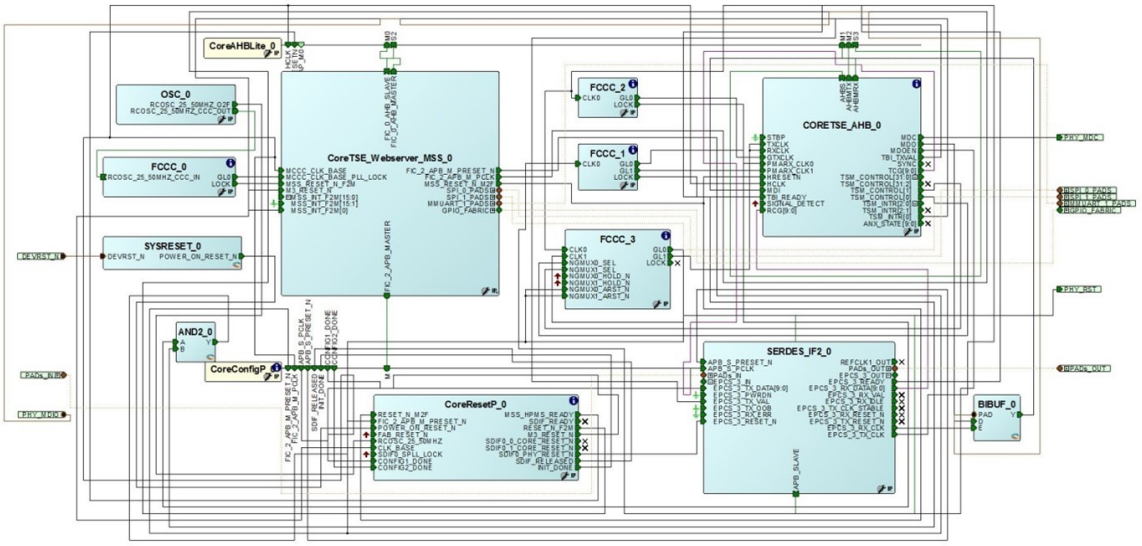
The following are the parameters:

- Fabric reset is used for CoreTSE_AHB reset.
- RXCLK is 125/25/2.5 MHz for 1000/100/10 Mbps respectively, and they are driven from Ethernet PHY
- TXCLK is 25/2.5 MHz for 100/10 Mbps respectively, and they are driven from Ethernet PHY
- GTXCLK is 125 MHz clock for 1000 Mbps and generated from on-board oscillator, same is supplied to the Ethernet PHY
- 50 MHz HCLK is required for the AHB subsystem in the core, which is driven from the application host clock.

Important: To run the Libero flow, enable **Timing Driven and High Effort Place and Route** option.

The following figure shows the CoreTSE_AHB integration in TBI system.

Figure 9-2. CoreTSE_AHB Integration in TBI System



The following are the parameters:

- Fabric reset is used for CoreTSE_AHB and SerDes resets
- PMARX_CLK0 and PMARX_CLK1 are 62.5 MHz and 180 degree out of phase clocks generated using FCCC_1. Input clock for the FCCC_1 is 125 MHz (EPCS_3_RX_CLK) from the SerDes.
- TXCLK and RXCLK are 2.5/25/125 MHz clocks generated from FCCC_3. The input for the FCCC_3 is 125 MHz from EPCS_3_TX_CLK from the SerDes.
- GTXCLK is 125 MHz clock generated from FCCC_2. The input for the FCCC_2 is 125 MHz from EPCS_3_TX_CLK from the SerDes.
- 50 MHz HCLK is required for the AHB subsystem in the core, which is driven from the application host clock.

Important: To run the Libero flow, enable **Timing Driven and High Effort Place and Route** option.

10. Ordering Codes (Ask a Question)

CoreTSE_AHB can be ordered through your local Microchip sales representative. It must be ordered using the following number scheme: CoreTSE_AHB -XX, where XX is listed in the following table.

Table 10-1. Ordering Codes

XX	Description
OM	Available as Obfuscated RTL source code

11. Additional References (Ask a Question)

This section provides additional references and information related to CoreTSE_AHB. For updates and additional information about the software, devices, and hardware, visit the **Intellectual Property** pages on the Microchip FPGAs and PLDs website.

11.1. Discontinued Features and Devices (Ask a Question)

There are no discontinued features and devices for CoreTSE_AHB v4.1.

11.2. Known Issues and Workarounds (Ask a Question)

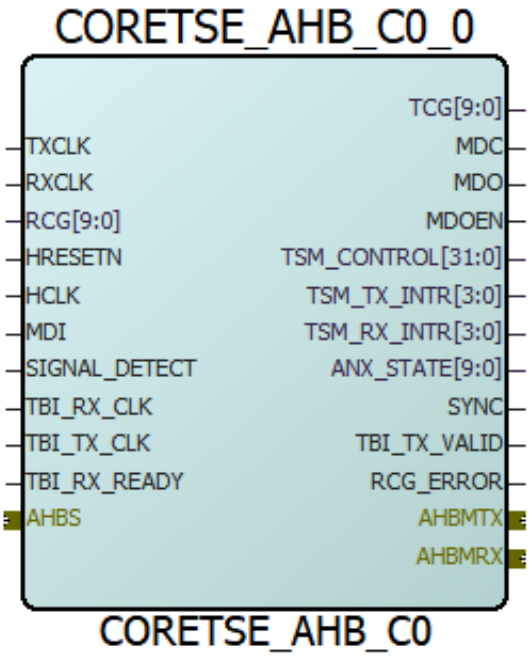
There are no known limitations and workarounds for CoreTSE_AHB v4.2.

11.3. Migration (Ask a Question)

The following list provides the process to migrate to CoreTSE_AHB v4.0 or later:

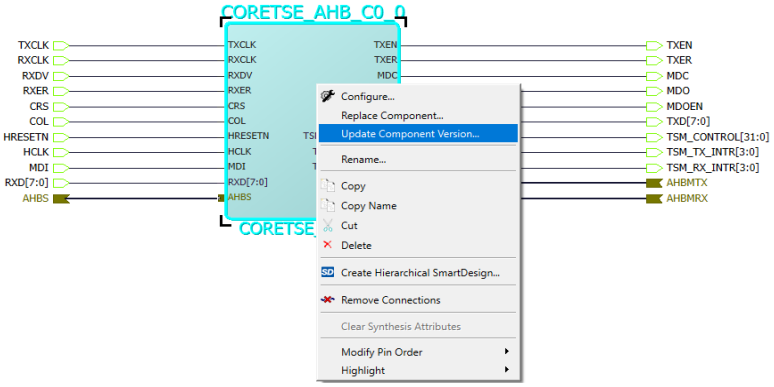
- STBP signal is now removed from the RTL, as it is always tied to Zero. This change does not impact any functionality.
- When migrating to CoreTSE_AHB v4.0 or later from earlier versions you can expect the following SmartDesign and logs windows.

Figure 11-1. Expected SmartDesign View when Migrating to CoreTSE_AHB v4.0 or Later



The following figure shows the updating SmartDesign to CoreTSE_AHB v4.0 or later using **Update Component Version** option.

Figure 11-2. Updating SmartDesign to CoreTSE_AHB v4.0 or Later using Update Component Version Option



The **Info** or **Error** in the following figure appears during the migration. You can safely ignore them as these are reported due to the newly added parameters "ECC_ENABLE", "TXRX_INTR_ENABLE", "TSM_TX_INTR" and "TSM_RX_INTR" ports.

Figure 11-3. Migration Errors

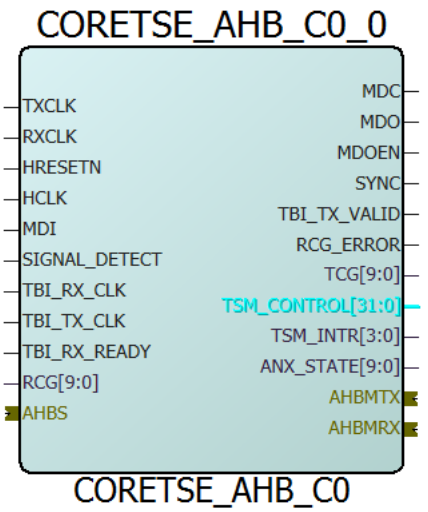
```
Error: Parameter 'ECC_ENABLE' not found in the spirit definition of instance 'CORETSE_AHB_C0_0'.
Error: Parameter 'TXRX_INTR_ENABLE' not found in the spirit definition of instance 'CORETSE_AHB_C0_0'.
Info: The connection is being dropped from the net 'TSM_TX_INTR' because the pin 'TSM_TX_INTR' does not exist.
Info: The connection is being dropped from the net 'TSM_RX_INTR' because the pin 'TSM_RX_INTR' does not exist.
Info: 'CORETSE_AHB_C0' was successfully generated.
Info: 'CORETSE_AHB_C0' manifest file 'C:/Prasanna/Libero/TSE_AHB_Migration/component/work/CORETSE_AHB_C0/
CORETSE_AHB_C0_manifest.txt' was successfully generated.

Info: Component 'CORETSE_AHB_C0' was successfully updated to 'CORETSE_AHB' core version '3.1.102'.
```

TSM_INTR Interrupt port is now 4-bit wide instead of 3-bit:

Changes: New bit is now added to the port for ECC interrupts.

- Bit [3]: Provides interrupt for Tx and Rx to SEC and DED events.
- Bit [2:0]: Provides the existing Interrupt functionality.

Figure 11-4. Common Interrupt Port Highlight



New Configurator Option to Enable Separate Interrupt Ports for Tx and Rx:

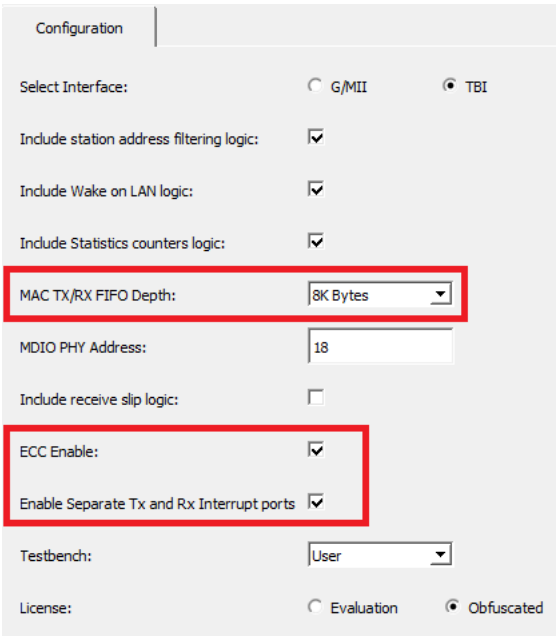
The separate interrupt ports, "TSM_TX_INTR" and "TSM_RX_INTR" are enabled by default, as shown in the following figure. You can always revert to common interrupt port that is, "TSM_INTR". To revert "TSM_INTR", in the **Configurator** window, uncheck the **Enable Separate Tx and Rx interrupt ports** option.

Figure 11-5. Separate Interrupt Ports Highlight

PACKET_SIZE option is now available as **MAC TX/RX FIFO Depth** in the configurator GUI.

The following figure shows the configurator window:

Figure 11-6. Configurator



12. Device Utilization and Performance (Ask a Question)

Device utilization and performance data are provided in the following tables for the SmartFusion® 2, IGLOO® 2 and PolarFire® devices. The data are indicative only. In TBI mode (1000 Mbps), TXCLK, RXCLK, TBI_TX_CLK and TBI_RX_CLK performance was above 125 MHz.

(TBI, MAC TX/RX FIFO Depth = 8 KB, include station address filtering logic = Yes, Include Wake on LAN logic = Yes, Include Statistics Counter logic = Yes, ECC Enable = Yes, and Enable Separate Tx and Rx Interrupt ports = Yes)

Important: To meet the timing, use the following instructions.

- **High Effort Layout** must be enabled in the Configure options section of **Place and Route** for all families before performing PnR.

Additionally, for RTG4™ family devices:

- Device speed grade must be –1.
- **Enable Single Event Transient Mitigation** option must be enabled in the **Project** > **Project Settings** > **Device settings**.
- MAC TX/RX FIFO Depth must be below 8 KB.

The following table lists the device utilization.

Table 12-1. CoreTSE_AHB Device Utilization

Device Details		Resources			Performance	RAMs
Family	Device	4LUT	DFF	Logic Elements		LSRAM
SmartFusion® 2	M2S010-1FG484	10102	5752	10737	HCLK= 109.290 MHz RXCLK= 154.823 MHz TXCLK= 176.929 MHz TBI_RX_CLK =180.278 MHz TBI_TX_CLK =183.925	14
IGLOO® 2	M2GL010-1FG484	9905	5630	10545	HCLK= 109.290 MHz RXCLK= 154.823 MHz TXCLK= 176.929 MHz TBI_RX_CLK= 180.278 MHz TBI_TX_CLK= 183.925 MHz	14
PolarFire®	MPF100T-1FCG484E	10791	6196	11304	HCLK= 143.699 MHz RXCLK= 277.239 MHz TXCLK= 196.040 MHz TBI_RX_CLK= 257.334 MHz TBI_TX_CLK= 290.192 MHz	24
RTG4™	RT4G150-1CG1657M	10018	5545	10742	HCLK= 94.913 MHz RXCLK= 135.999 MHz TXCLK=139.082 MHz TBI_RX_CLK= 142.694 MHz TBI_TX_CLK=131.857 MHz	7

Table 12-1. CoreTSE_AHB Device Utilization (continued)

Device Details		Resources			Performance	RAMs
Family	Device	4LUT	DFF	Logic Elements		LSRAM
PolarFire® SoC	MPFS095T-1FCSG536E	9990	5596	10592	HCLK = 175.377 MHz RXCLK = 296.560 MHz TXCLK = 315.159 MHz TBI_RX_CLK = 268.745 MHz TBI_TX_CLK = 355.745 MHz	12
RT PolarFire®	MPF500TS-1FCG784T2	9988	5587	10554	HCLK = 166.945 MHz RXCLK = 272.628 MHz TXCLK = 294.204 MHz TBI_RX_CLK = 276.778 MHz TBI_TX_CLK = 371.333 MHz	12

Table 12-2. (G/MII, PACKET_SIZE = 256 Bytes, SAL-OFF, WoL-OFF, STATS- OFF)

Family	FPGA Resources			Utilization	
	4LUT	DFF	Total	Device	%
SmartFusion® 2	4398	2821	7219	M2S150T	4.78%
IGLOO® 2	4398	2821	7219	M2GL150T	4.78%
PolarFire®	4319	2798	7117	MPF300TPES	2.4%

Table 12-3. (G/MII, PACKET_SIZE = 32K Bytes, SAL-ON, WoL-ON, STATS-ON)

Family	FPGA Resources			Utilization	
	4LUT	DFF	Total	Device	%
SmartFusion® 2	9570	6023	15593	M2S150T	10.4%
IGLOO® 2	9570	6023	15593	M2GL150T	10.4%
PolarFire®	9386	5979	15365	MPF300TPES	5.27%

Table 12-4. (TBI, PACKET_SIZE = 256 Bytes, SAL- OFF, WoL- OFF, STATS- OFF)

Family	FPGA Resources			Utilization	
	4LUT	DFF	Total	Device	%
SmartFusion® 2	6445	3863	10308	M2S150T	6.79%
IGLOO® 2	6445	3863	10308	M2GL150T	6.79%
PolarFire®	6366	3829	10195	MPF300TPES	3.48%

Important: Data in the preceding table is achieved using synthesis and layout settings optimized for speed.

13. Glossary (Ask a Question)

Following are the list of terms and definitions used in this document.

Table 13-1. Terms and Definitions

Term	Definition
AHB	AMBA High-performance Bus
AN	Auto Negotiation
DED	Double-bit Error Detection
DMA	Direct Memory Access
GMII	Gigabit Media Independent Interface
MII	Media Independent Interface
RX	Receiver
SEC	Single-bit Error Correction
TBI	Ten Bit Interface
TSE	Triple Speed Ethernet
TX	Transmitter

14. Resolved Issues (Ask a Question)

The following table lists the resolved issues.

Table 14-1. Resolved Issues

Version	Changes
4.2	The following is the list of changes in the v4.2 release: • Case Number: 01559798 An option is added to filter FCS bytes on the RX interface. • Case Number: 01311949 Updated CAR1 and CAR2 register access types and updated the width mismatch for the registers: • RBYT, RFCS, RXCF, RFLR and TNCL • Case Number: 01333337 Updated Register "0x010 - Maximum_frame_length" description • Updated Register Configuration Sequence for Frame Filtering and Pause Frame Transmission.
4.1	The following is the list of changes in the v4.1 release: • Resolved issue with MDIO not reading the PHY address correctly. • Updated the missing details for the bits [30:21] in AHB-DMA Descriptor Packet Information register. • Updated information for Simulation, Synthesis, Place and Route tool flow.
4.0	The following is the list of changes in the v4.0 release: • Added ECC feature with interrupt support • Timing constraints section was added in the user guide • Ports list, Parameters, and Register Maps sections were updated in the user guide
3.2	The following is the list of changes in the v3.2 release: • Added Self Destruct evaluation support • Resolved issue with CRC enabled for RTG4™ • Added SGMII interface in features and SGMII/TBI refer to the same interface
3.1	The following is the list of changes in the v3.1 release: • A typographical error in the Handbook utilization table was identified and corrected. • The default value for the MAC-FIFO Configuration Register 5, as stated in the handbook, was found to be mismatched with the actual default value. This has been rectified. • In the PolarFire® configuration, the RX_SLIP output port was added, and the barrel shift word aligner was removed.
3.0	The following is the list of changes in the v3.0 release: • Support for a single 125MHz TBI receive clock (PMA_RX_CLK). • Added support for PolarFire.
2.1	The following is the list of changes in the v2.1 release: • Added SYNC and ANX_STATE debug ports. • Added SIGNAL_DETECT port. • Resolved M-SGMII Link-Up Stability Performance Issue.
2.0	First production release

15. Revision History (Ask a Question)

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

Table 15-1. Revision History

Revision	Date	Description
D	02/2026	The following is the list of changes made in revision D of the document: • Updated the changes related to v4.2. • Updated Functional Description section as follows: – Added MAC Register Configuration for Pause Frames section. – Added Required Registers for Unicast Frame Filtering, Required Registers for Multicast and Broadcast Frame Filtering, and Required Registers for Filtering of Different Types of Error Frames sections in the Frame Filtering section. – Added Recommended Sequence to Clear CARRY Registers CAR1 and CAR2 Along With Stats Counters section in the Statistics Counters Logic section. • Updated Register Map and Descriptions section. • Updated Table 14-1 in the Resolved Issues section.
C	01/2025	The following is the list of changes made in revision C of the document: • Updated the changes related to v4.1. • Updated Table 3-2 in the Descriptor Information section. • Updated Table 4-1, Table 4-2, Table 4-3, and Table 4-4 in the CoreTSE_AHB Parameters and Interface Signals section. • Updated all the configuration GUI figures in the Implementing CoreTSE_AHB in Libero Design Suite section • Updated Timing Constraints section. • Updated User Testbench section with default configuration information.
B	07/2024	Updated Table 14-1 with version 4.0 changes
A	06/2024	The following is the list of changes made in revision A of the document: • The document was migrated to the Microchip template • The document number was updated to DS50003719A from 50200571 • Updated document for CoreTSE_AHB v4.0: – Added support for RTG4 family, see CoreTSE_AHB Summary section. – Added ECC feature with interrupt support. – Updated CoreTSE_AHB Parameters and Interface Signals section. – Added Timing Constraints section. – Updated Register Map and Descriptions section.
5.0	01/2018	Updated changes related to CoreTSE_AHB v3.2.
4.0	02/2017	Updated changes related to CoreTSE_AHB v3.1.
3.0	02/2016	Updated changes related to CoreTSE_AHB v3.0.
2.0	03/2015	Updated changes related to CoreTSE_AHB v2.1.
1.0	08/2014	Initial Release.

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-2805-4

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.