Mobility M6
Register Reference Guide

Technical Reference Manual

P/N: RRG-216M6-00-01OEM

The information contained in this manual has been carefully checked and is believed to be entirely reliable. No responsibility is assumed for inaccuracies. ATI reserves the right to make changes at any time to improve design and supply the best product possible.

Record of Revisions

Release	Date
RRG-216M6-00-01OEM	Aug 2000
Refer to Appendix B for details	

Related Manuals

Mobility M6 series

- Mobility M6 Graphics Controller Specifications
  CHS-216M6-00-01

Table of Contents

Appendix A: Cross Referenced Index

Appendix B: Revision History

Chapter 1
Introduction

1.1 About this Manual

This manual serves as a register reference guide to the M6 graphics controller.

- Chapter 1 outlines the notations and conventions used throughout this manual.
- Chapter 2 provides a summary of the Register Groups.
- Chapter 3 provides a detailed decription of the registers.
- Appendix A provides a cross-referenced list (sorted by Register Name and Address).

1.2 Nomenclature and Conventions

1.2.1 Register and Field Names

Mnemonics in upper-case are used throughout this document to represent hardware register names and field names. The naming conventions for registers and bit fields are as indicated below:

- REGISTER_MNEMONIC

  For example, CONFIG_CHIP_ID is the mnemonic for the Configuration Chip ID register.
- REGISTER_MNEMONIC[Bit_Numbers] or FIELD_NAME@REGISTER_MNEMONIC

  For example, CONFIG_CHIP_ID[15:0] refers to the bit field that occupies bit positions 0 through 15 within this register, whereas CFG_CHIP_TYPE@CONFIG_CHIP_ID gives the field name CFG_CHIP_TYPE (Product Type Code) instead of the bits position.

1.2.2 Grouped Registers

Registers that share the same descriptions are grouped and noted.

1.2.3 Numeric Representations

- Hexadecimal numbers are appended with “h” whenever there is a risk of ambiguity. Other numbers are assumed to be in decimal.
- Registers (or fields) of identical function are sometimes indicated by a single expression in which the part of the signal name that differs is enclosed in [ ] brackets. For example, the eight Host Data registers — HOST_DATA0 through to HOST_DATA7 — are represented by the single expression HOST_DATA[7:0].

1.2.4 Register Description

All registers in this document are described with the format of the self-explained sample table below. All offsets are in hexadecimal notation, while programmed bits are in either binomial or hexadecimal notation. (Note: sometimes not shown are the indirect type of byte offsets, e.g., CFG, PLL, etc., which will be indicated on the appropriate registers).

1.2.5 Typical Register Format and Explanation

MPEG_COPYRIGHT_NUMBER_1 MMR:0x0ED4 IND:0x0ED4 [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
COPYRIGHT_NUMBER_1	19:0	0x0	Stores bits 44 to 63 of the copyright_number.
(reserved)	31:20		

Table 1-1 Explanation of Format

Register Variable	Example
Register Name	MPEG-COPYRIGHT_NUMBER_1
Read / Write Capability R = Readable W = Writable	[R]
Register Size	32 bits
Accessibility by Bit Size	(access: 32)
Register Addresses	MMR:0x0ED4 IND:0x0ED4
Field Name	COPYRIGHT_NUMBER_1
Field position/size	19:0
Field description	Stores bits 44 to 63 of the copyright_number.l

1.3 Acronyms

Standard acronyms or abbreviations used in the literature are presumed known and therefore freely used without any explanation. When in doubt, refer to Table 1-1 below for a quick check. Less frequently used or ATI-specific acronyms will be accompanied by the full expression when appearing for the first time in the document.

Table 1-2 Acronyms

Acronym	Meaning
AGP	Accelerated Graphics Port
AMC	ATI Multimedia Channel
BIOS	basic input/output system
bpp	bits per pixel
CCE	Concurrent Command Engine
DAC	digital-to-analog converter
EDO RAM	Extended Data Output RAM
FIFO	first in first out
GUI	graphical user interface
I^2C	inter IC's communication
I/O	input/output
MPEG	Motion Picture Experts Group
MPP	Multimedia Peripheral Port
PCI	Peripheral Component Interconnect
PLL	phase-locked loop
POST	power-on self-test
RAMDAC	RAM digital-to-analog converter
RGB	red-green-blue (may refer to a color encoding scheme or a video signal)
R/W	read/write
SDRAM	Synchronous DRAM
SGRAM	Synchronous Graphics RAM
VGA	Video Graphics Array
VIP	Video Interface Port
WRAM	Windows RAM
YUV	A color encoding scheme, no direct correspondence to the letters

This page intentionally left blank.

Chapter 2
Registers Description

2.1 HostInterface Registers

2.2 cfg Registers

DEVICE_ID - R - 16 bits - [MMReg:0xF02] [pciConfig:0x2]			
Field Name	**Bits**	**Default**	**Description**
DEVICE_ID	15:0	0x5159	

No description available for this register.

VENDOR_ID - R - 16 bits - [MMReg:0xF00] [pciConfig:0x0]			
Field Name	**Bits**	**Default**	**Description**
VENDOR_ID	15:0	0x1002	

No description available for this register.

COMMAND - RW - 16 bits - [MMReg:0xF04]:R [pciConfig:0x4]			
Field Name	**Bits**	**Default**	**Description**
IO_ACCESS_EN	0	0x0	0=Disable 1=Enable
MEM_ACCESS_EN	1	0x0	0=Disable 1=Enable
BUS_MASTER_EN	2	0x0	0=Disable 1=Enable
SPECIAL_CYCLE_EN (R)	3	0x0	0=Disable 1=Enable
MEM_WRITE_INVALIDATE_EN (R)	4	0x0	0=Disable 1=Enable
PAL_SNOOP_EN	5	0x0	0=Disable 1=Enable
PARITY_ERROR_EN (R)	6	0x0	0=Disable 1=Enable
AD_STEPPING (R)	7	0x1	0=Disable 1=Enable
SERR_EN	8	0x0	0=Disable 1=Enable
FAST_B2B_EN	9	0x0	0=Disable 1=Enable

No description available for this register.

STATUS - RW - 16 bits - [MMReg:0xF06]:R [pciConfig:0x6]			
Field Name	**Bits**	**Default**	**Description**
CAP_LIST (R)	4	0x1	
PCI_66_EN (R)	5	0x1	
UDF_EN (R)	6	0x0	0=Disable 1=Enable
FAST_BACK_CAPABLE (R)	7	0x1	
DEVSEL_TIMING (R)	10:9	0x1	
SIGNAL_TARGET_ABORT (R)	11	0x0	
RECEIVED_TARGET_ABORT	12	0x0	0=Inactive 1=Active
RECEIVED_MASTER_ABORT	13	0x0	0=Inactive 1=Active
SIGNALED_SYSTEM_ERROR	14	0x0	
PARITY_ERROR_DETECTED (R)	15	0x0	

No description available for this register.

REVISION_ID - R - 8 bits - [MMReg:0xF08] [pciConfig:0x8]			
Field Name	**Bits**	**Default**	**Description**
MINOR_REV_ID	3:0	0x0	
MAJOR_REV_ID	7:4	0x0	

No description available for this register.

IO_BASE - RW - 32 bits - [MMReg:0xF14]:R [pciConfig:0x14]			
Field Name	**Bits**	**Default**	**Description**
BLOCK_IO_BIT (R)	7:0	0x1	
IO_BASE	31:8	0x0	

No description available for this register.

REG_BASE - RW - 32 bits - [MMReg:0xF18]:R [pciConfig:0x18]			
Field Name	**Bits**	**Default**	**Description**
REG_BASE	31:16	0x0	

No description available for this register.

MEM_BASE - RW - 32 bits - [MMReg:0xF10]:R [pciConfig:0x10]			
Field Name	**Bits**	**Default**	**Description**
PREFETCH_EN (R)	3	0x1	
MEM_BASE	31:25	0x0	

No description available for this register.

ADAPTER_ID_W - RW - 32 bits - [MMReg:0xF4C]:R [pciConfig:0x4C]			
Field Name	**Bits**	**Default**	**Description**
SUBSYSTEM_VENDOR_ID	15:0	0x0	
SUBSYSTEM_ID	31:16	0x0	

No description available for this register.

BASE_CODE - R - 8 bits - [MMReg:0xF0B] [pciConfig:0xB]			
Field Name	**Bits**	**Default**	**Description**
BASE_CLASS_CODE	7:0	0x3	

No description available for this register.

ADAPTER_ID - R - 32 bits - [MMReg:0xF2C] [pciConfig:0x2C]			
Field Name	**Bits**	**Default**	**Description**
SUBSYSTEM_VENDOR_ID *(mirror of ADAPTER_ID_W:SUBSYSTEM_VENDOR_ID)*	15:0	0x0	
SUBSYSTEM_ID *(mirror of ADAPTER_ID_W:SUBSYSTEM_ID)*	31:16	0x0	

No description available for this register.

BIOS_ROM - RW - 32 bits - [MMReg:0xF30]:R [pciConfig:0x30]			
Field Name	**Bits**	**Default**	**Description**
BIOS_ROM_EN	0	0x0	0=Disable 1=Enable
BIOS_BASE_ADDR	31:17	0x0	

No description available for this register.

SUB_CLASS - R - 8 bits - [MMReg:0xF0A] [pciConfig:0xA]			
Field Name	**Bits**	**Default**	**Description**
SUB_CLASS_INF	7	0x0	0=00 - VGA device 1=80 - extended graphics

No description available for this register.

BIST - R - 8 bits - [MMReg:0xF0F] [pciConfig:0xF]			
Field Name	**Bits**	**Default**	**Description**
BIST_COMP	3:0	0x0	
BIST_STRT	6	0x0	
BIST_CAP	7	0x0	

No description available for this register.

CAPABILITIES_PTR - R - 32 bits - [MMReg:0xF34] [pciConfig:0x34]			
Field Name	**Bits**	**Default**	**Description**
CAP_PTR	7:0	0x50	

No description available for this register.

CONFIG_CNTL - RW - 32 bits - [IOReg,MMReg:0xE0]			
Field Name	**Bits**	**Default**	**Description**
APER_REG_ENDIAN	5:4	0x0	
CFG_VGA_RAM_EN *(mirror of GENMO_WT:VGA_RAM_EN)*	8	0x0	0=Disable 1=Enable
CFG_VGA_IO_DIS	9	0x0	
CFG_ATI_REV_ID (R)	19:16	0x0	

No description available for this register.

CONFIG_MEMSIZE - RW - 32 bits - [IOReg,MMReg:0xF8]			
Field Name	**Bits**	**Default**	**Description**
CONFIG_MEMSIZE	28:24	0x0	

No description available for this register.

CONFIG_APER_0_BASE - R - 32 bits - [MMReg:0x100]			
Field Name	**Bits**	**Default**	**Description**
APER_0_BASE	31:25	0x0	

No description available for this register.

CONFIG_APER_1_BASE - R - 32 bits - [MMReg:0x104]			
Field Name	**Bits**	**Default**	**Description**
APER_1_BASE	31:24	0x0	

No description available for this register.

CONFIG_APER_SIZE - R - 32 bits - [MMReg:0x108]			
Field Name	**Bits**	**Default**	**Description**
APER_SIZE	27:24	0x0	

No description available for this register.

CONFIG_REG_1_BASE - R - 32 bits - [MMReg:0x10C]			
Field Name	**Bits**	**Default**	**Description**
REG_APER_1_SELECT	15	0x1	
REG_1_BASE	31:16	0x0	

No description available for this register.

CONFIG_REG_APER_SIZE - R - 32 bits - [MMReg:0x110]			
Field Name	**Bits**	**Default**	**Description**
REG_APER_SIZE	15:0	0x8000	

No description available for this register.

HEADER - R - 8 bits - [MMReg:0xF0E] [pciConfig:0xE]			
Field Name	**Bits**	**Default**	**Description**
HEADER_TYPE	6:0	0x0	
DEVICE_TYPE	7	0x0	0=Single-Function Device 1=Multi-Function Device

No description available for this register.

INTERRUPT_LINE - RW - 8 bits - [MMReg:0xF3C]:R [pciConfig:0x3C]			
Field Name	**Bits**	**Default**	**Description**
INTERRUPT_LINE	7:0	0xff	

No description available for this register.

INTERRUPT_PIN - R - 8 bits - [MMReg:0xF3D] [pciConfig:0x3D]			
Field Name	**Bits**	**Default**	**Description**
INTERRUPT_PIN	0	0x0	

No description available for this register.

LATENCY - RW - 8 bits - [MMReg:0xF0D]:R [pciConfig:0xD]			
Field Name	**Bits**	**Default**	**Description**
LATENCY_TIMER	7:0	0x0	

No description available for this register.

MAX_LATENCY - R - 8 bits - [MMReg:0xF3F] [pciConfig:0x3F]			
Field Name	**Bits**	**Default**	**Description**
MAX_LAT	7:0	0x0	

No description available for this register.

REGPROG_INF - R - 8 bits - [MMReg:0xF09] [pciConfig:0x9]			
Field Name	**Bits**	**Default**	**Description**
REG_LEVEL_PROG_INF	7:0	0x0	

No description available for this register.

CACHE_LINE - RW - 8 bits - [MMReg:0xF0C]:R [pciConfig:0xC]			
Field Name	**Bits**	**Default**	**Description**
CACHE_LINE_SIZE	7:0	0x0	

No description available for this register.

MIN_GRANT - R - 8 bits - [MMReg:0xF3E] [pciConfig:0x3E]			
Field Name	**Bits**	**Default**	**Description**
MIN_GNT	7:0	0x8	

Miscellaneous Output Register (Write)

2.3 busControl Registers

BUS_CNTL - RW - 32 bits - [IOReg,MMReg:0x30]			
Field Name	**Bits**	**Default**	**Description**
BUS_DBL_RESYNC	0	0x1	0=Normal 1=Add extra resynchronizing clock
BUS_MSTR_RESET (W)	1	0x0	0=Normal 1=Reset
BUS_FLUSH_BUF (W)	2	0x0	0=Normal 1=Flush
BUS_STOP_REQ_DIS	3	0x0	0=Normal 1=Disable
BUS_READ_COMBINE_EN	4	0x0	0=Disable 1=Enable
BUS_WRT_COMBINE_EN	5	0x0	0=Disable 1=Enable
BUS_MASTER_DIS	6	0x1	0=Enable 1=Disable
BIOS_ROM_WRT_EN	7	0x0	0=Disable 1=Enable
BUS_PREFETCH_MODE	9:8	0x0	0=HI prefetching disabled 1=HI prefetches during active bus requests only 2=HI always prefetches continuously 3=Hi attempts to hold 1 DW of read data for a delayed read completion
BUS_VGA_PREFETCH_EN	10	0x0	0=Disable 1=Enable
BUS_SGL_READ_DISABLE	11	0x0	0=Enable 1=Disable
BIOS_DIS_ROM	12	0x0	0=Enable 1=Disable
BUS_PCI_READ_RETRY_EN	13	0x0	0=Normal 1=Enable
BUS_AGP_AD_STEPPING_EN	14	0x1	0=No stepping in AGP 1=AD Stepping in AGP and PCI
BUS_PCI_WRT_RETRY_EN	15	0x0	0=Normal 1=Enable
BUS_RETRY_WS	19:16	0xf	

BUS_MSTR_RD_MULT	20	0x0	0=Read line 1=Read multiple
BUS_MSTR_RD_LINE	21	0x0	0=Read multiple 1=Read line
BUS_SUSPEND	22	0x0	0=Resume BM transfer 1=Suspend BM transfer
LAT_16X	23	0x0	0=1X 1=16X
BUS_RD_DISCARD_EN	24	0x0	0=Disable 1=Enable
ENFRCWRDY	25	0x0	0=Disable 1=Enable
BUS_MSTR_WS	26	0x0	0=8 wait states 1=32 wait states
BUS_PARKING_DIS	27	0x1	0=Enable 1=Disable
BUS_MSTR_DISCONNECT_EN	28	0x0	0=Disable 1=Enable
SERR_EN	29	0x0	0=Disable 1=Enable
BUS_READ_BURST	30	0x0	0=Disable 1=Enable
BUS_RDY_READ_DLY	31	0x1	0=no RDY delay 1=RDY delayed 1 mem clk

No description available for this register.

BUS_CNTL1 - RW - 32 bits - [IOReg,MMReg:0x34]			
Field Name	**Bits**	**Default**	**Description**
PMI_IO_DISABLE	0	0x0	0=Normal 1=Disable
PMI_MEM_DISABLE	1	0x0	0=Normal 1=Disable
PMI_BM_DISABLE	2	0x0	0=Normal 1=Disable
PMI_INT_DISABLE	3	0x0	0=Normal 1=Disable
BUS2_STALE_DATA_TIMER	7:4	0xf	

BUS2_VGA_REG_COHERENCY_DIS	8	0x0	0=Enable 1=Disable
BUS2_VGA_MEM_COHERENCY_DIS	9	0x0	0=Enable 1=Disable
BUS2_HDP_REG_COHERENCY_DIS	10	0x0	0=Enable 1=Disable
BUS2_GUI_INITIATOR_COHERENCY_DIS	11	0x0	0=Enable 1=Disable
MOBILE_PLATFORM_SEL	27:26	0x0	graphic chip selects which mobile platform it is supporting 0=Do not choose any mobile platform 1=Solano2-M platform 2=440BX platform 3=Do not choose any mobile platform
SEND_SBA_LATENCY	30:28	0x0	defines delay to send SBA again after STP_AGP# is de-asserted 0=50us 1=60us 2=70us 3=80us 4=100us 5=200us 6=400us 7=800us
AGPCLK_VALID	31	0x0	defines which signal indicates AGP clock is valid 0=AGP clock do not stop 1=use SUS_STAT# to indicate AGP clock is valid

Miscellaneous Output Register (Write)

HI_STAT - RW - 32 bits - [IOReg,MMReg:0x4C]			
Field Name	**Bits**	**Default**	**Description**
HI_STAT	2:0	0x0	
AGP_BUSY (R)	3	0x0	

No description available for this register.

BM_STATUS - R - 32 bits - [MMReg:0x160]			
Field Name	**Bits**	**Default**	**Description**
BUS_MASTER_STATUS	31:0	0x0	

No description available for this register.

AGP_COMMAND - RW - 32 bits - [MMReg:0xF60]:R [pciConfig:0x60]			
Field Name	**Bits**	**Default**	**Description**
DATA_RATE	2:0	0x0	
AGP_EN	8	0x0	0=disable 1=enable
SBA_EN (R)	9	0x1	0=Disable 1=Enable
RQ_DEPTH	31:24	0x0	

No description available for this register.

AGP_CNTL - RW - 32 bits - [MMReg:0x174]			
Field Name	**Bits**	**Default**	**Description**
MAX_IDLE_CLK	7:0	0x0	
HOLD_RD_FIFO	8	0x0	
HOLD_RQ_FIFO	9	0x0	
EN_2X_STBB	10	0x0	
FORCE_FULL_SBA	11	0x0	
SBA_DIS	12	0x0	
AGP_REV_ID	13	0x0	
REG_CRIPPLE_AGP4X	14	0x0	
REG_CRIPPLE_AGP2X4X	15	0x0	
FORCE_INT_VREF	16	0x0	
PENDING_SLOTS_VAL	18:17	0x0	
PENDING_SLOTS_SEL	19	0x0	
EN_EXTENDED_AD_STB_2X	20	0x0	
DIS_QUEUED_GNT_FIX	21	0x0	0=Enable 1=Disable
EN_RDATA2X4X_MULTIRESET	22	0x0	
EN_RBFCALM	23	0x0	

FORCE_EXT_VREF	24	0x0	
DIS_RBF	25	0x0	0=Allow normal RBF operation 1=Disable RBF generation
AGP_MISC	31:26	0x0	

No description available for this register.

AGP_CAP_ID - R - 32 bits - [MMReg:0xF58] [pciConfig:0x58]			
Field Name	**Bits**	**Default**	**Description**
CAP_ID	7:0	0x2	
NEXT_PTR	15:8	0x50	
AGP_MINOR	19:16	0x0	
AGP_MAJOR	23:20	0x2	

No description available for this register.

AGP_STATUS - R - 32 bits - [MMReg:0xF5C] [pciConfig:0x5C]			
Field Name	**Bits**	**Default**	**Description**
RATE1X	0	0x1	
RATE2X	1	0x1	
RATE4X	2	0x1	
SBA	9	0x1	
RQ	31:24	0x2f	

No description available for this register.

MM_INDEX - RW - 32 bits - [IOReg,MMReg:0x0]			
Field Name	**Bits**	**Default**	**Description**
MM_ADDR	30:0	0x0	
MM_APER	31	0x0	0=Register Aperture 1=Linear Aperture 0

No description available for this register.

MM_DATA - RW - 32 bits - [IOReg,MMReg:0x4]			
Field Name	**Bits**	**Default**	**Description**
MM_DATA	31:0	0x0	

No description available for this register.

PAD_CTLR_STRENGTH - RW - 32 bits - [MMReg:0x168]			
Field Name	**Bits**	**Default**	**Description**
PAD_N_STRENGTH_READ_BACK (R)	3:0	0x0	
PAD_P_STRENGTH_READ_BACK (R)	7:4	0x0	
PAD_N_MANUAL_STRENGTH	11:8	0x0	
PAD_P_MANUAL_STRENGTH	15:12	0x0	
PAD_MANUAL_OVERRIDE	16	0x1	0=Allow normal impedance compensation operation 1=Default to manual settings
PAD_TEST_OUT (R)	17	0x0	
PAD_DUMMY_OUT (R)	18	0x0	
PAD_HI_IO_DFR	19	0x0	
PAD_HI_IO_SCHMEN	20	0x0	
PAD_HI_IO_DREN	21	0x0	
PAD_HI_IO_SLEW	22	0x0	
PAD_HI_IO_VDIFF	23	0x0	
PAD_HI_IO_DFR_OVERRIDE	24	0x0	
PAD_HI_IO_SCHMEN_OVERRIDE	25	0x0	
PAD_HI_IO_DREN_OVERRIDE	26	0x0	
PAD_HI_IO_SLEW_OVERRIDE	27	0x0	
PAD_HI_IO_VDIFF_OVERRIDE	28	0x0	

No description available for this register.

PAD_CTLR_UPDATE - RW - 32 bits - [MMReg:0x16C]			
Field Name	**Bits**	**Default**	**Description**
PAD_UPDATE_RATE	4:0	0x16	
PAD_SAMPLE_DELAY	12:8	0x6	
PAD_INC_THRESHOLD	20:16	0x18	
PAD_DEC_THRESHOLD	28:24	0x8	

No description available for this register.

PAD_AGPINPUT_DELAY - RW - 32 bits - [MMReg:0x164]			
Field Name	**Bits**	**Default**	**Description**
PAD_AGPINPUT_DELAY	31:0	0x0	

No description available for this register.

GENENB - R - 8 bits - VGA_IO:0x3C3			
Field Name	**Bits**	**Default**	**Description**
BLK_IO_BASE	7:0	0x0	

No description available for this register.

GENMO_WT - W - 8 bits - VGA_IO:0x3C2			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B (BIF)	0	0x0	VGA addressing mode. 0=Monochrome emulation, regs at 0x3Bx 1=Color/Graphic emulation, regs at 0x3Dx
VGA_RAM_EN (BIF)	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture. 0=Disable 1=Enable

VGA_CKSEL (DISPLAY)	3:2	0x0	Selects pixel clock frequency to use in VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=0. See CLOCK_CNTL_INDEX.PPLL_DIV_SEL for non-VGA mode pixel clock selection. 0=25.1744MHz (640 Pels) 1=28.3212MHz (720 Pels) 2=Reserved 3=Reserved
ODD_EVEN_MD_PGSEL (HDP)	5	0x0	This bit is used in odd/even display modes (A/N modes: 0, 1, 2, 3, and 7). This bit is ignored when either bit GRA06[1] or SEQ4[3] are enabled. Used to determine if the VGA aperture maps into the lower (even) or upper (odd) page of memory. 0=Selects odd (high) memory locations 1=Selects even (low) memory locations
VGA_HSYNC_POL (DISPLAY)	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low The covention of VGA is to use active low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL (DISPLAY)	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low The covention of VGA is to use active high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

Miscellaneous Output Register (Write)

GENMO_RD - R - 8 bits - VGA_IO:0x3CC			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B *(mirror of GENMO_WT:GENMO_MONO_ADDRESS_B)* (BIF)	0	0x0	VGA addressing mode.
VGA_RAM_EN *(mirror of GENMO_WT:VGA_RAM_EN)* (BIF)	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture.

VGA_CKSEL *(mirror of GENMO_WT:VGA_CKSEL)* (DISPLAY)	3:2	0x0	Selects pixel clock frequency to use.
ODD_EVEN_MD_PGSEL *(mirror of GENMO_WT:ODD_EVEN_MD_PGSEL)* (HDP)	5	0x0	This bit is used in odd/even display modes (A/N modes: 0, 1, 2, 3, and 7). This bit is ignored when either bit GRA06[1] or SEQ4[3] are enabled. Used to determine if the VGA aperture maps into the lower (even) or upper (odd) page of memory.
VGA_HSYNC_POL *(mirror of GENMO_WT:VGA_HSYNC_POL)* (DISPLAY)	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low The covention of VGA is to use active low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL *(mirror of GENMO_WT:VGA_VSYNC_POL)* (DISPLAY)	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low The covention of VGA is to use active high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

Miscellaneous Output Register (Read)

DAC_CNTL - RW - 32 bits - [IOReg,MMReg:0x58]			
Field Name	**Bits**	**Default**	**Description**
DAC_RANGE_CNTL (DISPLAY)	1:0	0x2	DAC control bits. Default = 2. 0=Reserved 1=Reserved 2=PS2 Output Level 3=YPbPr output level
DAC_BLANKING (DISPLAY)	2	0x0	Controls use of DAC blanking pedestal during horizontal and vertical blanks. VGA PS2 compatible monitors expect a 0 IRE blanking pedestal.

DAC_CMP_EN (DISPLAY)	3	0x1	Control DAC comparators for analog termination checking. When enabled, the results of the three comparators are read back in the DAC_CMP_OUT_R/G/B fields. Use for the DAC_FORCE fields of DAC_EXT_CNTL is recommended for testing analog monitor connection. DAC_FORCE allows the correct 10 bit data values to be forced on the DAC channels without corrupting the TMDS or TV out images. When using the DAC comparators, be sure the comparator settling times are met by waiting at least 1us between changing the DAC_FORCE parameters and reading the comparator status bits. The comparator refence voltage is 0.440V. The recommended 10 bit DAC value for testing 75 or 37.5 ohm termination for PS2 current levels is 0x1AC. For each channel, if the comparator output is 0, it indictates the termination is 75 ohms, and therefore no monitor is attached. When the comparator output is 1, the termination is 37.5 ohms and a monitor is attached.
DAC_CMP_OUT_R (R) (DISPLAY)	4	0x0	Red channel comparator output.
DAC_CMP_OUT_G (R) (DISPLAY)	5	0x0	Green channel comparator output.
DAC_CMP_OUT_B (R) (DISPLAY)	6	0x0	Blue channel comparator output.
DAC_CMP_OUTPUT (R) (DISPLAY)	7	0x0	Logical AND of R, G & B comparator outputs.
DAC_8BIT_EN (DISPLAY)	8	0x0	Enables 8 bit DAC operation. 8 bit is normal, 6 bit used for VGA emulation. When in 6 bit writes and reads to DAC_DATA and PALETTE_DATA are affected. Writes shift 6 bits left by 2 to make 8 bits in the palette memory. Reads shift 8 bit palette data right by 2 to give 6 MSBs to the host. 0=DAC_DATA and PALETTE_DATA read/writes emulate 6 bit palette 1=DAC_DATA and PALETTE_DATA read/writes emulate 8 bit palette
DAC_4BPP_PIX_ORDER (DISPLAY)	9	0x0	Selects the order of pixel nibbles within bytes for 4 bpp extended (non-VGA) display modes.
DAC_VGA_ADR_EN (BIF)	13	0x0	Enables access of the palette (DAC) at the VGA I/O DAC addresses when in extended display modes (non-VGA, or CRTC_EXT_DISP_EN=1).

DAC_EXPAND_MODE (DISPLAY)	14	0x0	Method to expand 4,5 or 6 bit Red/Green/Blue color codes to 8 bit addresses. 0 = Zero Extension 1 = Dynamic Expansion 0=Convert 4, 5 & 6 bit colors to 8 bits by zero extension 1=Convert 4, 5 & 6 bit colors to 8 bits by dynamic expansion
DAC_PDWN (DISPLAY)	15	0x0	Power down internal DAC macro bandgap unit. Should also set the R, G & B powerdowns in the DAC_MACRO_CNTL register at the same time. Powering down the DAC does not affect the digital outputs (TV or flat panel). The DAC is automatically powered down when the PMI_POWER_STATE register is not in the D0 state. Setting all DAC_PDWN fields should save about 56 mA when PS2 output levels.
CRT_SENSE (R) (DISPLAY)	16	0x0	0=CRT Monitor Detection result - not connected 1=CRT Monitor Detection result - connected
CRT_DETECTION_ON (DISPLAY)	17	0x0	0=CRT Monitor Detection disabled 1=CRT Monitor Detection enabled
DAC_CRC_CONT_EN (DISPLAY)	18	0x0	When CRC is in one shot mode, one-and-only-one frame or field is CRCed after DAC_CRC_EN is set high. In continuous mode every frame/field is CRCed and the results are valid for one field/frame until the next set of results are ready. If a field/frames results are not read within one frame in continuous mode they are over written. In one-shot mode the results remain and can be read until DAC_CRC_EN is cleared. 0=DAC CRC runs in one shot mode. 1=DAC CRC runs in continuous mode.
DAC_CRC_EN (DISPLAY)	19	0x0	Enables the CRC signature check on the data going to the DAC macro. This is what appears on the screen, and includes graphics, HW cursor, video overlay, sub-picture, and overscan. CRC will start in next vertical blank on the first pixel of the line where VSYNC goes active, and run for one field/frame. For interlaced modes the CRC block will not start working until the beginning of a field with odd/even polarity matching the DAC_CRC_FIELD setting. 0=Reset DAC CRC. 1=Enable DAC CRC for next frame/field in one shot, or all frames/fields in continuous mode.
DAC_CRC_FIELD (DISPLAY)	20	0x0	Used only for interlaced mode CRCs. Controls which field polarity starts the CRC block after DAC_CRC_EN is set high. 0=If interlace display and one shot mode, then CRC even field only 1=If interlace display and one shot mode, then CRC odd field only

DAC_LUT_COUNTER_LIMIT (DISPLAY)	22:21	0x0	Anti-sparkle timeout. The palette circuit will search for a place in the display image to hide palette reads or writes. If no hiding location is found within this many pixels, a dot-strectch is forced to hide the cycle as best as possible. Setting this register too high in low resolution modes may cause long delays on the PCI/AGP bus. The recommended settings are best. 0=anti-sparkle timeout 3 clocks 1=anti-sparkle timeout 7 clocks 2=anti-sparkle timeout 15 clocks 3=anti-sparkle timeout 31 clocks
DAC_LUT_READ_SEL (DISPLAY)	23	0x0	Used for diagnostics only. Selects palette for HOST reads. 0=Palette reads come from main palette 1=Palette reads come from secondary palette
DAC_MASK (DISPLAY)	31:24	0xff	Mirror VGA DAC_MASK. No affect in non-VGA modes. Masks off usage of individual palette index bits before pixel index is looked-up in the palette. 0 = do not use this bit of the index 1 = use this bit of the index This is a mirror of the VGA DAC_MASK register. It only has an effect in VGA emulation modes (CRTC_EXT_DISP_EN=0), not for VESA modes or extended display modes.

General control for the RGB DAC and palette.

CRTC_GEN_CNTL - RW - 32 bits - [IOReg,MMReg:0x50]			
Field Name	**Bits**	**Default**	**Description**
CRTC_DBL_SCAN_EN (DISPLAY)	0	0x0	Double scan enable. Double scan only affects the calculation of display addresses by adding the CRTC_PITCH every second line (also applies to the hardware cursor pitch). Enabling double scan does not change the CRTC vertical programming or VSYNC timing. The overscan top & bottom are not affected and remain the number of lines programmed (i.e. not double). The hardware cursor programming remains in terms of physical lines (not logical lines). The cursor vertical position must begin on an even line number when in double scan. The cursor itself is limited to 64 physical lines in height, which means only 32 logical lines. This is because the cursor pitch is only added at the end of odd scan lines, but the CRTC vertical logic stops the cursor after 64 physical lines. 0=disable 1=enable
CRTC_INTERLACE_EN (DISPLAY)	1	0x0	Interlace display mode enable. 0=Non-Interlace 1=Interlace

CRTC_C_SYNC_EN (DISPLAY)	4	0x0	Enables composite sync on horizontal sync output. When this is set, the VSYNC pin should be disabled by setting CRTC_EXT_CNTL.CRTC_VSYNC_TRISTATE=1. 0=Disable 1=Enable
CRTC_PIX_WIDTH (DISPLAY)	11:8	0x0	Display pixel width (actually depth). For 4bpp mode DAC_CNTL.DAC_4BPP_PIX_ORDER selects the order of nibbles within bytes. When R, G, or B components are only 4, 5 or 6 bits, DAC_CNTL.DAC_EXPAND_MODE selects how these components are expanded to 8 bits each for keying and palette lookup. When alpha values are 1 or 4 bits, they are expanded to 8 bits by dynamic expansion of the high order bits to the missing lower order bits. 0=Disable pixel clock for primary CRTC 1=4bpp Indexed 2=8bpp Indexed 3=15bpp aRGB 1555 4=16bpp RGB 565 5=24bpp RGB 888 6=32bpp aRGB 8888 7=16bpp aRGB 4444 8=16bpp aIndex 88
CRTC_ICON_EN (DISPLAY)	15	0x0	0=Disable Hardware Icon 1=Enable Hardware Icon
CRTC_CUR_EN (DISPLAY)	16	0x0	Hardware cursor enable. This field is double buffered and locked with the CUR_LOCK register field. 0=Disable 1=Enable

CRTC_VSTAT_MODE (DISPLAY)	18:17	0x0	Selects the location of the display updating of CRTC_OFFSET and CUR_OFFSET and related fields during the vertical retrace. Also determines where the VGA_VSTATUS update will occur within the vertical retrace for VGA modes. For non-VGA modes only the upper bit of this field is used. When 00 or 01 in non-VGA the updating of the CRTC_OFFSET and the CUR_OFFSET is delayed as long as possible within the vertical retrace until the start of the last line of the retrace. When 10 or 11 in non-VGA the CRTC_OFFSET and CUR_OFFSET are updated at the leading edge of VSYNC, which is normally relatively early in the vertical retrace. No matter how this is set, the update of status bits like WAIT_UNTIL_PFLIP will reflect the actual location of the update. Any writes to CRTC_OFFSET or CUR_OFFSET after the selected point has passed in the vertical retrace will have no affect until the display frame after the next vertical retrace. For VGA modes this field affects the behaviour of page flipping in some applications/games that poll the VGA_VSTATUS flag. This field should be tuned by the BIOS for compatibility with the most games. 0=VGA_VSTATUS until vcount= vtotal, DISP_ADDR loads when vcount=vt otal 1=VGA_VSTATUS until vblank end, DISP_ADDR loads when vcount=vtotal 2=VGA_VSTATUS until vcount= vtotal,DISP_ADDR loads in vsync start 3=VGA_VSTATUS until vblank end,DISP_ADDR loads in vsync start
CRTC_CUR_MODE (DISPLAY)	22:20	0x0	Hardware cursor mode. For 2bpp mode, each line of cursor data is stored in memory as 64 bits of AND data followed by 64 bits or XOR data. For color AND/XOR mode, each pixel is stored sequentially in memory as 32bits each in aRGB8888 format with bit 31 of each DWord being the AND bit. For the color alpha modes the format is also 32bpp aRGB8888 with all 8 bits of the alpha being used. All HW cursor lines must be 64 pixels wide and all lines must be stored sequentially in memory. 0=Mono + 1 bit AND (2bpp), 64x64 1=Color 24bpp + 1 bit AND, 32hx64v 2=Color 24bpp + 8 bit aplha, premultiplied alpha, 32hx64v 3=Color 24bpp + 8 bit aplha, unmultiplied alpha, 32hx64v
CRTC_EXT_DISP_EN (BIF) (DISPLAY)	24	0x0	Extended display mode enable. No affect if strapped VGA_DISABLE=1. 0=VGA 1=Extended

CRTC_EN (DISPLAY)	25	0x0	Enables CRT controller. When reset, the CRTC horizontal counter is set to zero, and the vertical counter is set to the current value of CRTC_V_TOTAL_DISP.CRTC_V_DISP. 0=Reset 1=Enable
CRTC_DISP_REQ_EN_B (DISPLAY)	26	0x1	Enables display graphics requests to the memory controller. Affects only graphics and VGA text requests. Does not affect HW cursor, overlay or subpic. When setting this bit, CRTC_EXT_CNTL.CRTC_DISPLAY_DIS is also normally set to blank the screen. Active low. 0=Enable 1=Disable

CRTC general controls

2.4 gart Registers

AIC_CTRL - RW - 32 bits - [MMReg:0x1D0]			
Field Name	**Bits**	**Default**	**Description**
TRANSLATE_EN	0	0x0	0=disable pci gart 1=enable pci gart
HW_0_DEBUG	1	0x0	
HW_1_DEBUG	2	0x0	
HW_2_DEBUG	3	0x0	
HW_3_DEBUG	4	0x0	
HW_4_DEBUG	5	0x0	
HW_5_DEBUG	6	0x0	
HW_6_DEBUG	7	0x0	
HW_7_DEBUG	8	0x0	
HW_8_DEBUG	9	0x0	
HW_9_DEBUG	10	0x0	
HW_A_DEBUG	11	0x0	
HW_B_DEBUG	12	0x0	
HW_C_DEBUG	13	0x0	
HW_D_DEBUG	14	0x0	
HW_E_DEBUG	15	0x0	
HW_F_DEBUG	16	0x0	
HW_10_DEBUG	17	0x0	
HW_11_DEBUG	18	0x0	
HW_12_DEBUG	19	0x0	
HW_13_DEBUG	20	0x0	
HW_14_DEBUG	21	0x0	
HW_15_DEBUG	22	0x0	
HW_16_DEBUG	23	0x0	
HW_17_DEBUG	24	0x0	

HW_18_DEBUG	25	0x0	
HW_19_DEBUG	26	0x0	
HW_1A_DEBUG	27	0x0	
HW_1B_DEBUG	28	0x0	
HW_1C_DEBUG	29	0x0	
HW_1D_DEBUG	30	0x0	
HW_1E_DEBUG	31	0x0	

PCI GART page table base address in system memory

AIC_STAT - R - 32 bits - [MMReg:0x1D4]			
Field Name	**Bits**	**Default**	**Description**
AIC_TLB_VLD	0	0x0	
AIC_STAT1	1	0x0	
AIC_STAT0	2	0x0	

PCI GART page table base address in system memory

AIC_PT_BASE - RW - 32 bits - [MMReg:0x1D8]			
Field Name	**Bits**	**Default**	**Description**
AIC_PT_BASE	31:12	0x0	

PCI GART page table base address in system memory

AIC_LO_ADDR - RW - 32 bits - [MMReg:0x1DC]			
Field Name	**Bits**	**Default**	**Description**
AIC_LO_ADDR	31:12	0x0	

PCI GART page table base address in system memory

AIC_HI_ADDR - RW - 32 bits - [MMReg:0x1E0]			
Field Name	**Bits**	**Default**	**Description**
AIC_HI_ADDR	31:12	0x0	

Upper boundary of PCI addresses responded to by the PCI GART; addresses higher than this are not translated

AIC_TLB_ADDR - R - 32 bits - [MMReg:0x1E4]			
Field Name	**Bits**	**Default**	**Description**
AIC_TLB_ADDR	31:12	0x0	

Current page address before translation by the PCI GART

AIC_TLB_DATA - R - 32 bits - [MMReg:0x1E8]			
Field Name	**Bits**	**Default**	**Description**
AIC_TLB_DATA	31:12	0x0	

Current page address after translation by the PCI GART

2.5 pmi Registers

PMI_CAP_ID - R - 8 bits - [MMReg:0xF50] [pciConfig:0x50]			
Field Name	**Bits**	**Default**	**Description**
PMI_CAP_ID	7:0	0x1	1=PCI Bus Power Management Interface (PMI) register section

No description available for this register.

PMI_NXT_CAP_PTR - R - 8 bits - [MMReg:0xF51] [pciConfig:0x51]			
Field Name	**Bits**	**Default**	**Description**
PMI_NXT_CAP_PTR	7:0	0x0	

No description available for this register.

PM_STATUS - RW - 16 bits - [MMReg:0xF54]:R [pciConfig:0x54]			
Field Name	**Bits**	**Default**	**Description**
PMI_POWER_STATE	1:0	0x0	
PMI_PME_EN (R)	8	0x0	
PMI_DATA_SELECT (R)	12:9	0x0	
PMI_DATA_SCALE (R)	14:13	0x0	
PMI_PME_STATUS (R)	15	0x0	

No description available for this register.

PMI_PMC_REG - R - 16 bits - [MMReg:0xF52] [pciConfig:0x52]			
Field Name	**Bits**	**Default**	**Description**
PMI_VERSION	2:0	0x2	2=Compliant with PMI Specification verison 1.1
PMI_PME_CLOCK	3	0x0	
PMI_DEV_SPECIFIC_INIT	5	0x0	
PMI_D1_SUPPORT	9	0x1	

PMI_D2_SUPPORT	10	0x1	
PMI_PME_SUPPORT	15:11	0x0	

No description available for this register.

PMI_DATA - R - 8 bits - [MMReg:0xF57] [pciConfig:0x57]			
Field Name	**Bits**	**Default**	**Description**
PMI_DATA	7:0	0x0	

No description available for this register.

2.6 clockGenerator Registers

CLK_PWRMGT_CNTL - RW - 32 bits - CLKIND:0x14			
Field Name	**Bits**	**Default**	**Description**
MPLL_PWRMGT_OFF (CGM)	0	0x0	M domain clock power management off
SPLL_PWRMGT_OFF (CG)	1	0x0	S domain clock power management off
PPLL_PWRMGT_OFF (CG)	2	0x0	Pixel clock power management off
P2PLL_PWRMGT_OFF (CG)	3	0x0	
MCLK_TURNOFF (CGM)	4	0x0	Turn off M domain clocks
SCLK_TURNOFF (CG)	5	0x0	Turn off S domain clocks
PCLK_TURNOFF (CG)	6	0x0	Turn off pixel clocks
P2CLK_TURNOFF (CG)	7	0x0	
TEST_MODE (CG) (CGM)	9	0x0	Disable long internal timing to speed up regression tests
GLOBAL_PMAN_EN (CG)	10	0x0	0=Power managment off 1=Power managment on
ENGINE_DYNCLK_MODE (CG)	12	0x1	0=Provide Clock for each Eng block separately 1=Treat Engine as one single block
ACTIVE_HILO_LAT (CG)	14:13	0x3	0=5 clocks 1=12 clocks 2=20 clocks 3=32 clocks

DISP_DYN_STOP_LAT (CG)	15	0x0	0=10 clocks 1=20 clocks
MC_BUSY (R) (CGM)	16	0x0	0=MC is idle 1=MC is busy
MC_INT_CNTL (CGM)	17	0x0	0=HW control 1=SW over-ride
MC_SWITCH (CGM)	18	0x0	0=source of memory clock is not changed 1=source of memory clock is changed
DLL_READY (CGM)	19	0x0	0=DLL is not ready 1=DLL is ready
DISP_PM (CG)	20	0x0	0=display clocks running in PM modes 1=clocks OFF
DYN_STOP_MODE (CG)	23:21	0x7	0=10 clocks 7=111 = 140 clocks
CG_NO1_DEBUG (CG) (CGM)	29:24	0x0	1st 16-bit hardware debug register
TVPLL_PWRMGT_OFF (CG)	30	0x0	
TVCLK_TURNOFF (CG)	31	0x0	

Clock power management control

PLL_PWRMGT_CNTL - RW - 32 bits - CLKIND:0x15			
Field Name	**Bits**	**Default**	**Description**
MPLL_TURNOFF (CGM)	0	0x0	Enable M domain PLL to be turned off at power state D3
SPLL_TURNOFF (CG)	1	0x0	Enable S domain PLL to be turned off at power state D3

PPLL_TURNOFF (CG)	2	0x0	Enable pixel clock PLL to be turned off at power state D3
P2PLL_TURNOFF (CG)	3	0x0	
TVPLL_TURNOFF (CG)	4	0x0	
AGPCLK_DYN_STOP_LAT (CG)	8:5	0x0	
APM_POWER_STATE (CG)	10:9	0x0	
APM_PWRSTATE_RD (R) (CG)	12:11	0x0	
PM_MODE_SEL (CG)	13	0x0	0=ACPI 1=APM
EN_PWRSEQ_DONE_COND (CG)	14	0x1	0=Switch states without PWRSEQ_DONE rising edge condition 1=Enable condition
EN_DISP_PARKED_COND (CG)	15	0x1	0=Switch states without DISP_PARKED condition 1=Enable condition
MOBILE_SU (CG)	16	0x1	0=Regular 1=Optimize power comsumption in Suspend mode
SU_SCLK_USE_BCLK (CG)	17	0x1	1=During Suspend, SCLK is sourced from BCLK
SU_MCLK_USE_BCLK (CG)	18	0x1	1=During Suspend, MCLK is sourced from BCLK
SU_SUSTAIN_DISABLE (CG)	19	0x0	0=Sustain Suspend until PLL lockup 1=Disable
TCL_BYPASS_DISABLE (CG)	20	0x0	0=Enable TCL_SCLK power management 1=Disable
TCL_CLOCK_ACTIVE_RD (R) (CG)	21	0x0	0=TCL_SCLK stopped 1=TCL_SCLK running

CG_NO2_DEBUG (CG) (CGM)	31:24	0x0	2nd 16-bit hardware debug register

PLL power management control

CLK_PIN_CNTL - RW - 32 bits - CLKIND:0x1			
Field Name	**Bits**	**Default**	**Description**
OSC_EN	0	0x1	Oscillation enable 0=Disable 1=Enable
XTL_LOW_GAIN	2	0x1	Oscillator high gain or low gain control 0=High Gain 1=Low Gain
DONT_USE_XTALIN	4	0x0	0=Use XTALIN pin to generate internal timing (for simulation only) 1=Use divided version of PLL reference clock
SLOW_CLOCK_SOURCE	5	0x0	0=XTALIN is used for power sequencing block 1=PCICLKBY2
CG_CLK_TO_OUTPIN	11	0x0	0=disabled 1=send out selected clock for jitter test
CG_COUNT_UP_TO_OUTPIN	12	0x0	0=disabled 1=send out COUNT_UP signal for test tunning
CG_SPARE	14:13	0x0	
SCLK_DYN_START_CNTL	15	0x1	0=SCLK starts 4 clocks after BUSY active 1=SCLK starts 1 clock after BUSY active
CG_SPARE_RD (R)	18:16	0x0	
XTALIN_ALWAYS_ONb	19	0x0	0=force XTALIN to run in Suspend mode 1=PM control over XTALIN pad
PWRSEQ_DELAY	31:24	0x0	0=Programmable value of panel power sequencing block. 1=This value can be programmed up to 225 ms in increments of 1 ms (generated from F32KHz clock). 2=If 0 is programmed, SLOW_CLOCK is stopped

Clock pin control

PPLL_CNTL - RW - 32 bits - CLKIND:0x2			
Field Name	**Bits**	**Default**	**Description**
PPLL_RESET	0	0x1	Power down pixel clock PLL 0=Not Reset 1=Reset
PPLL_SLEEP	1	0x1	Reset pxiel clock PLL 1=Powerdown
PPLL_TST_EN	2	0x0	Enable pixel clock PLL test mode
PPLL_REFCLK_SEL	4	0x0	0=not flopped 1=flopped
PPLL_FBCLK_SEL	5	0x0	0=not flopped 1=flopped
PPLL_TCPOFF	6	0x0	0=normal 1=test mode: Hi-Z
PPLL_TVCOMAX	7	0x0	0=normal 1=test mode: runaway
PPLL_PCP	10:8	0x4	Pixel clock PLL charge pump gain. Programmed by BIOS. Do not change.
PPLL_PVG	13:11	0x4	Pixel clock PLL VCO gain. Programmed by BIOS. Do not change.
PPLL_PDC	15:14	0x1	Pixel clock PLL duty cycle. Programmed by BIOS. Do not change.
PPLL_ATOMIC_UPDATE_EN	16	0x0	Pixel clock PLL atomic update enable for non-VGA modes. Reference and feedback dividers are double buffered and updated concurrently when a PPLL_ATOMIC_UPDATE_W field is set. 0=Atomic Update Disabled 1=Atomic Update Enabled
PPLL_VGA_ATOMIC_UPDATE_EN	17	0x0	Same as PPLL_ATOMIC_UPDATE_EN, but for VGA modes. 0=VGA Atomic Update Disabled 1=VGA Atomic Update Enabled
PPLL_ATOMIC_UPDATE_SYNC	18	0x0	Selects pixel clock PLL atomic update trigger position. Using VSYNC delays update, but allows clock frequency change to be hidden in the vertical retrace. 0=Update ASAP 1=Update in VSYNC

Pixel clock PLL control

PPLL_REF_DIV - RW - 32 bits - CLKIND:0x3			
Field Name	**Bits**	**Default**	**Description**
PPLL_REF_DIV	9:0	0x3f	Pixel clock PLL reference divider. Must be set so reference frequency is between 1MHz and 400kHz. i.e. 1MHz >= (PPLL_REF) / PPLL_REF_DIV >= 400 kHz. Also 0 and 1 and not legal settings for this register field.
PPLL_ATOMIC_UPDATE_W (W)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers. 0=No Update 1=Update
PPLL_ATOMIC_UPDATE_R (R)	15	0x0	Readback of atomic update status. 0=Update done 1=Update Pending
PPLL_REF_DIV_SRC	17:16	0x0	Pixel clock PLL reference divider input source selection. 0=PPLL_REF = XTALIN 1=PPLL_REF = PLLSCLK/2 2=PPLL_REF = PLLSCLK/4 3=PPLL_REF = Secondary Reference Clock

Pixel clock PLL reference division configuration

SPLL_CNTL - RW - 32 bits - CLKIND:0xC			
Field Name	**Bits**	**Default**	**Description**
SPLL_SLEEP	0	0x1	Power down S domain PLL 1=Powerdown
SPLL_RESET	1	0x1	Reset S domain PLL 0=Not Reset 1=Reset
SPLL_TST_EN	2	0x0	Enable S domain PLL test mode
SPLL_REFCLK_SEL	4	0x0	0=not flopped 1=flopped
SPLL_FBCLK_SEL	5	0x0	0=not flopped 1=flopped
SPLL_TCPOFF	6	0x0	0=normal 1=test mode: Hi-Z
SPLL_TVCOMAX	7	0x0	0=normal 1=test mode: runaway
SPLL_PCP	10:8	0x4	Program S domain PLL charge pump
SPLL_PVG	13:11	0x4	Program S domain PLL VCO gain

SPLL_PDC	15:14	0x1	Program S domain clock duty cycle
SPLL_X1_CLK_SKEW	18:16	0x0	
SPLL_X2_CLK_SKEW	22:20	0x0	
SPLL_MODE	27:24	0x4	
MYCLK_SOURCED_FROM_SPLL_SEL	29:28	0x0	0=SPLLBY1 1=SPLLBY1b 2=SPLLBY2 3=SPLLBY2b
ENABLE_MYCLK_FROM_SPLL	30	0x0	0=clock stop 1=clock run

S domain PLL control

SCLK_CNTL - RW - 32 bits - CLKIND:0xD			
Field Name	**Bits**	**Default**	**Description**
SCLK_SRC_SEL	2:0	0x0	S domain clock source selection 0=not PCICLK 1=SCLK/1 2=SCLK/2 3=SCLK/4 4=SCLK/8 5=External Source/Scan Clock (AUXWIN pin) 6=Test Clock from Test Controller 7=MPLL
CP_MAX_DYN_STOP_LAT	3	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
HDP_MAX_DYN_STOP_LAT	4	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
TV_MAX_DYN_STOP_LAT	5	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
E2_MAX_DYN_STOP_LAT	6	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
SE_MAX_DYN_STOP_LAT	7	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
IDCT_MAX_DYN_STOP_LAT	8	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
VIP_MAX_DYN_STOP_LAT	9	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
RE_MAX_DYN_STOP_LAT	10	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency

PB_MAX_DYN_STOP_LAT	11	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
TAM_MAX_DYN_STOP_LAT	12	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
TDM_MAX_DYN_STOP_LAT	13	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
RB_MAX_DYN_STOP_LAT	14	0x1	0=Programmable dynamic stopping latency 1=Max dynamic stopping latency
FORCE_DISP2	15	0x1	0=Dynamic 1=ForceOn
FORCE_CP	16	0x1	CP block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_HDP	17	0x1	HDP block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_DISP1	18	0x1	0=Dynamic 1=ForceOn
FORCE_TOP	19	0x1	0=Dynamic 1=ForceOn
FORCE_E2	20	0x1	E2 block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_SE	21	0x1	SE block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_IDCT	22	0x1	IDCT block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_VIP	23	0x1	VIP block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_RE	24	0x1	RE block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_PB	25	0x1	PB block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_TAM	26	0x1	TAM block dynamic power management off 0=Dynamic 1=ForceOn

FORCE_TDM	27	0x1	TDM block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_RB	28	0x1	RB block dynamic power management off 0=Dynamic 1=ForceOn
FORCE_TV_SCLK	29	0x1	0=Dynamic 1=ForceOn
FORCE_SUBPIC	30	0x1	0=Dynamic 1=ForceOn
FORCE_OV0	31	0x1	0=Dynamic 1=ForceOn

Clock source selection and dynamic power management control

AGP_PLL_CNTL - RW - 32 bits - CLKIND:0xB			
Field Name	**Bits**	**Default**	**Description**
APLL_SLEEP	0	0x0	Power down AGP PLL
APLL_RESET	1	0x0	Reset AGP PLL
APLL_XSEL	3:2	0x0	Select different AGP mode for testing 0=TESTMODE REFCLK 1=TESTMODE AGP1X 2=TESTMODE AGP2X 3=TESTMODE AGP4X
APLL_TST_EN	4	0x0	Enable AGP PLL test mode 0=normal 1=test mode
APLL_TCPOFF	5	0x0	0=normal 1=test mode: Hi-Z
APLL_TVCOMAX	6	0x0	0=normal 1=test mode: runaway
APLL_REF_SKEW	9:7	0x4	
APLL_FB_SKEW	12:10	0x0	
APLL_X0_CLK_SKEW	15:13	0x0	
APLL_X1_CLK_SKEW	18:16	0x0	Program AGP1X mode clock skew
APLL_X2_CLK_SKEW	21:19	0x0	Program AGP2X mode clock skew
APLL_X4_CLK_SKEW	24:22	0x0	Program AGP4X mode clock skew
APLL_PUMP_GAIN	27:25	0x4	Program AGP PLL charge pump

APLL_VCO_GAIN	30:28	0x5	

AGP PLL control

TV_PLL_FINE_CNTL - RW - 32 bits - CLKIND:0x20			
Field Name	**Bits**	**Default**	**Description**
TV_M1	10:0	0x0	
TV_N1	21:11	0x0	
TV_DIVIDER_SEL	22	0x0	0=M0,N0 1=M1,N1
TV_MNFLIP_REQ	23	0x0	
TV_MNFLIP_DONE (R)	24	0x0	0=M/N Update In Progress 1=M/N Update Done
TV_SLIP_REQ	25	0x0	
TV_SLIP_DONE (R)	26	0x0	0=Slipping In Progress 1=Slipping Done
TV_MNFLIP_EN	27	0x0	0=Disable 1=Enable
TV_SLIP_COUNT	31:28	0x1	

No description available for this register.

TV_PLL_CNTL - RW - 32 bits - CLKIND:0x21			
Field Name	**Bits**	**Default**	**Description**
TV_M0_LO	7:0	0x39	
TV_N0_LO	16:8	0xf9	
TV_M0_HI	20:18	0x0	
TV_N0_HI	22:21	0x0	
TV_SLIP_EN	23	0x0	0=Disable 1=Enable
TV_P	27:24	0x3	
TV_DTO_EN	28	0x0	0=Disable 1=Enable

TV_DTO_TYPE	29	0x0	0=Autoslip 1=Automatic M/N Flip
TV_REF_CLK_SEL	30	0x0	0=Reference Clock 1=Secondary Reference Clock

No description available for this register.

TV_PLL_CNTL1 - RW - 32 bits - CLKIND:0x22			
Field Name	**Bits**	**Default**	**Description**
TVPLL_RESET	1	0x1	0=Run 1=Reset
TVPLL_SLEEP	3	0x1	1=Powerdown TVPLL
TVPLL_REFCLK_SEL	4	0x0	0=not flopped 1=flopped
TVPLL_FBCLK_SEL	5	0x0	0=not flopped 1=flopped
TVPLL_TCPOFF	6	0x0	0=normal 1=test mode: Hi-Z
TVPLL_TVCOMAX	7	0x0	0=normal 1=test mode: runaway
TVPCP	10:8	0x4	
TVPVG	13:11	0x4	
TVPDC	15:14	0x1	
TVCLK_SRC_SEL	30	0x0	0=CPUCLK 1=TVPLLCLK
TVPLL_TEST	31	0x1	0=Test PLL during reset mode if TEST_PLL_EN=1 1=No PLL Testing

Clock pin control

TV_DTO_INCREMENTS - RW - 32 bits - CLKIND:0x23			
Field Name	**Bits**	**Default**	**Description**
TV_DTO_INC0	15:0	0x0	
TV_DTO_INC1	31:16	0x0	

No description available for this register.

P2PLL_CNTL - RW - 32 bits - CLKIND:0x2A			
Field Name	**Bits**	**Default**	**Description**
P2PLL_RESET	0	0x1	0=Not Reset P2PLL 1=Reset P2PLL
P2PLL_SLEEP	1	0x1	1=Powerdown P2PLL
P2PLL_TST_EN	2	0x0	
P2PLL_REFCLK_SEL	4	0x0	0=not flopped 1=flopped
P2PLL_FBCLK_SEL	5	0x0	0=not flopped 1=flopped
P2PLL_TCPOFF	6	0x0	0=normal 1=test mode: Hi-Z
P2PLL_TVCOMAX	7	0x0	0=normal 1=test mode: runaway
P2PLL_PCP	10:8	0x4	
P2PLL_PVG	13:11	0x4	
P2PLL_PDC	15:14	0x1	
P2PLL_ATOMIC_UPDATE_EN	16	0x0	0=Atomic Update Disabled 1=Atomic Update Enabled
P2PLL_ATOMIC_UPDATE_SYNC	18	0x0	0=Update ASAP 1=Update in VSYNC

No description available for this register.

P2PLL_REF_DIV - RW - 32 bits - CLKIND:0x2B			
Field Name	**Bits**	**Default**	**Description**
P2PLL_REF_DIV	9:0	0x0	
P2PLL_ATOMIC_UPDATE_W (W)	15	0x0	0=No Update 1=Update
P2PLL_ATOMIC_UPDATE_R (R)	15	0x0	0=Update done 1=Update Pending
P2PLL_REF_DIV_SRC	17:16	0x0	0=P2PLL_REF = XTALIN 1=P2PLL_REF = PLLSCLK/2 2=P2PLL_REF = PLLSCLK/4 3=P2PLL_REF = Secondary Reference Clock

No description available for this register.

CLOCK_CNTL_INDEX - RW - 32 bits - [IOReg,MMReg:0x8]			
Field Name	**Bits**	**Default**	**Description**
PLL_ADDR (DISPLAY) (CG) (CGM)	5:0	0x0	Indirect CG and PLL register index
PLL_WR_EN (DISPLAY) (CG) (CGM)	7	0x0	Indirect CG and PLL register write enable 0=Disable writes to CLOCK_CNTL_DATA 1=Enable writing to CLOCK_CNTL_DATA
PPLL_DIV_SEL (DISPLAY) (CG) (CGM)	9:8	0x0	Pixel clock PLL feedback division selection for non-VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=1. See GENMO_WT.VGA_CKSEL for pixel clock selection when VGA mode active. 0=PPLL_DIV0 1=PPLL_DIV1 2=PPLL_DIV2 3=PPLL_DIV3

CG and PLL indirect register control

CLOCK_CNTL_DATA - RW - 32 bits - [IOReg,MMReg:0xC]			
Field Name	**Bits**	**Default**	**Description**
PLL_DATA (DISPLAY) (CG) (CGM)	31:0	0x0	Data for write to CG and PLL indirect registers

Data for write to CG and PLL indirect registers

PPLL_DIV_0 - RW - 32 bits - CLKIND:0x4			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB0_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.

PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST0_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 0

PPLL_DIV_1 - RW - 32 bits - CLKIND:0x5			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB1_DIV (CG)	10:0	0x1f2	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST1_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 1

PPLL_DIV_2 - RW - 32 bits - CLKIND:0x6			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB2_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST2_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 2

PPLL_DIV_3 - RW - 32 bits - CLKIND:0x7			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB3_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.

PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST3_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 3

HTOTAL_CNTL - RW - 32 bits - CLKIND:0x9			
Field Name	**Bits**	**Default**	**Description**
HTOT_PIX_SLIP (DISPLAY)	3:0	0x0	Pixel accurate control of horizontal total. Selects the extra number of pixels to add to each display line. Valid range is 0 to 15. For VGA modes with SEQ_PCLKBY2 = 1 each increment adds one physical pixel or 1/2 a logical pixel to the line total. For 9-dot 40 column VGA text modes, it is not possible to add 16/18ths or 17/18ths of a character extra to the display line times.
HTOT_VCLK_SLIP (DISPLAY)	11:8	0x0	Reserved for future use. No affect in this ASIC.
HTOT_PPLL_SLIP (CG)	18:16	0x0	Select the number of 1/5 PPllClk phase slips to do in the PLL at every HSYNC. This is used as a sub-pixel accurate adjustment of the frame rate for TV out or video gen-locking.
HTOT_CNTL_EDGE (DISPLAY) (CG)	24	0x0	Select which HSYNC edge the slip correction based on HTOT_PPLL_SLIP is done.
HTOT_CNTL_VGA_EN (DISPLAY) (CG)	28	0x0	Select if the slip controls based on HTOT_PIX_SLIP, HTOT_VCLK_SLIP and HTOT_PPLL_SLIP are enabled for VGA mode. These are always enabled for non-VGA modes when the respective fields are non-zero.

Horizontal total control. Used to fine-tune the horizontal total. This lengthens the time of each display line by sub-character and/or sub-pixel amounts. The purpose is fine adjustment of the overall frame refresh rate for applications that require it (e.g. TV output, GEN-lock to video input).

M_SPLL_REF_FB_DIV - RW - 32 bits - CLKIND:0xA			
Field Name	**Bits**	**Default**	**Description**
M_SPLL_REF_DIV (CG) (CGM)	7:0	0x0	Memory/System PLL reference division
MPLL_FB_DIV (CGM)	15:8	0x0	Memory PLL feedback division
SPLL_FB_DIV (CG)	23:16	0x0	System PLL feedback division
MPLL_REF_SRC_SEL (CGM)	24	0x0	0=Normal 1=Cleaner one through I/O

Memory/System PLL reference/feedback division configuration

PLL_TEST_CNTL - RW - 32 bits - CLKIND:0x13			
Field Name	**Bits**	**Default**	**Description**
TST_SRC_SEL (CG) (CGM)	6:0	0x0	Select different clock source to readable counter and as test clock output
TST_SRC_INV (CG) (CGM)	7	0x0	Invert test clock output
TST_DIVIDERS (CG) (CGM)	8	0x0	Enable driver test
PLL_MASK_READ_B (CG) (CGM)	9	0x1	Mask test clock output and 3 LSBs of readable counter
TESTCLK_MUX_SEL (CG)	12	0x0	0=cg 1=cgm

ANALOG_MON (DISPLAY)	23:15	0x0	
TEST_COUNT (R) (CG) (CGM)	31:24	0x0	Readable counter read back

CG and PLL test control

P2PLL_DIV_0 - RW - 32 bits - CLKIND:0x2C			
Field Name	**Bits**	**Default**	**Description**
P2PLL_FB_DIV (CG)	10:0	0x0	
P2PLL_ATOMIC_UPDATE_W *(mirror of P2PLL_REF_DIV:P2PLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	
P2PLL_ATOMIC_UPDATE_R *(mirror of P2PLL_REF_DIV:P2PLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	
P2PLL_POST_DIV (DISPLAY)	18:16	0x0	0=V2CLK = V2CLK_SRC 1=V2CLK = V2CLK_SRC/2 2=V2CLK = V2CLK_SRC/4 3=V2CLK = V2CLK_SRC/8 4=V2CLK = V2CLK_SRC/3 5=reserved 6=V2CLK = V2CLK_SRC/6 7=V2CLK = V2CLK_SRC/12

No description available for this register.

HTOTAL2_CNTL - RW - 32 bits - CLKIND:0x2E			
Field Name	**Bits**	**Default**	**Description**
HTOT2_PIX_SLIP (DISPLAY)	3:0	0x0	
HTOT2_PIX2CLK_SLIP (DISPLAY)	11:8	0x0	
HTOT2_P2PLL_SLIP (CG)	18:16	0x0	
HTOT2_CNTL_EDGE (DISPLAY) (CG)	24	0x0	
HTOT2_CNTL_UPDATE (CG)	28	0x0	

No description available for this register.

2.7 memClockGenerator Registers

CLK_PWRMGT_CNTL - RW - 32 bits - CLKIND:0x14			
Field Name	**Bits**	**Default**	**Description**
MPLL_PWRMGT_OFF (CGM)	0	0x0	M domain clock power management off
SPLL_PWRMGT_OFF (CG)	1	0x0	S domain clock power management off
PPLL_PWRMGT_OFF (CG)	2	0x0	Pixel clock power management off
P2PLL_PWRMGT_OFF (CG)	3	0x0	
MCLK_TURNOFF (CGM)	4	0x0	Turn off M domain clocks
SCLK_TURNOFF (CG)	5	0x0	Turn off S domain clocks
PCLK_TURNOFF (CG)	6	0x0	Turn off pixel clocks
P2CLK_TURNOFF (CG)	7	0x0	
TEST_MODE (CG) (CGM)	9	0x0	Disable long internal timing to speed up regression tests
GLOBAL_PMAN_EN (CG)	10	0x0	0=Power managment off 1=Power managment on
ENGINE_DYNCLK_MODE (CG)	12	0x1	0=Provide Clock for each Eng block separately 1=Treat Engine as one single block
ACTIVE_HILO_LAT (CG)	14:13	0x3	0=5 clocks 1=12 clocks 2=20 clocks 3=32 clocks

DISP_DYN_STOP_LAT (CG)	15	0x0	0=10 clocks 1=20 clocks
MC_BUSY (R) (CGM)	16	0x0	0=MC is idle 1=MC is busy
MC_INT_CNTL (CGM)	17	0x0	0=HW control 1=SW over-ride
MC_SWITCH (CGM)	18	0x0	0=source of memory clock is not changed 1=source of memory clock is changed
DLL_READY (CGM)	19	0x0	0=DLL is not ready 1=DLL is ready
DISP_PM (CG)	20	0x0	0=display clocks running in PM modes 1=clocks OFF
DYN_STOP_MODE (CG)	23:21	0x7	0=10 clocks 7=111 = 140 clocks
CG_NO1_DEBUG (CG) (CGM)	29:24	0x0	1st 16-bit hardware debug register
TVPLL_PWRMGT_OFF (CG)	30	0x0	
TVCLK_TURNOFF (CG)	31	0x0	

Clock power management control

PLL_PWRMGT_CNTL - RW - 32 bits - CLKIND:0x15			
Field Name	**Bits**	**Default**	**Description**
MPLL_TURNOFF (CGM)	0	0x0	Enable M domain PLL to be turned off at power state D3
SPLL_TURNOFF (CG)	1	0x0	Enable S domain PLL to be turned off at power state D3

PPLL_TURNOFF (CG)	2	0x0	Enable pixel clock PLL to be turned off at power state D3
P2PLL_TURNOFF (CG)	3	0x0	
TVPLL_TURNOFF (CG)	4	0x0	
AGPCLK_DYN_STOP_LAT (CG)	8:5	0x0	
APM_POWER_STATE (CG)	10:9	0x0	
APM_PWRSTATE_RD (R) (CG)	12:11	0x0	
PM_MODE_SEL (CG)	13	0x0	0=ACPI 1=APM
EN_PWRSEQ_DONE_COND (CG)	14	0x1	0=Switch states without PWRSEQ_DONE rising edge condition 1=Enable condition
EN_DISP_PARKED_COND (CG)	15	0x1	0=Switch states without DISP_PARKED condition 1=Enable condition
MOBILE_SU (CG)	16	0x1	0=Regular 1=Optimize power comsumption in Suspend mode
SU_SCLK_USE_BCLK (CG)	17	0x1	1=During Suspend, SCLK is sourced from BCLK
SU_MCLK_USE_BCLK (CG)	18	0x1	1=During Suspend, MCLK is sourced from BCLK
SU_SUSTAIN_DISABLE (CG)	19	0x0	0=Sustain Suspend until PLL lockup 1=Disable
TCL_BYPASS_DISABLE (CG)	20	0x0	0=Enable TCL_SCLK power management 1=Disable

TCL_CLOCK_ACTIVE_RD (R) (CG)	21	0x0	0=TCL_SCLK stopped 1=TCL_SCLK running
CG_NO2_DEBUG (CG) (CGM)	31:24	0x0	2nd 16-bit hardware debug register

PLL power management control

MPLL_CNTL - RW - 32 bits - CLKIND:0xE			
Field Name	**Bits**	**Default**	**Description**
MPLL_RESET	0	0x1	Reset M domain PLL 0=Not Reset 1=Reset
MPLL_SLEEP	1	0x1	Power down M domain PLL 1=Powerdown
MPLL_TST_EN	2	0x0	Enable M domain PLL test mode
MPLL_REFCLK_SEL	4	0x0	0=not flopped 1=flopped
MPLL_FBCLK_SEL	5	0x0	0=not flopped 1=flopped
MPLL_TCPOFF	6	0x0	0=normal 1=test mode: Hi-Z
MPLL_TVCOMAX	7	0x0	0=normal 1=test mode: runaway
MPLL_PCP	10:8	0x4	Program M domain PLL charge pump
MPLL_PVG	13:11	0x4	Program M domain PLL VCO gain
MPLL_PDC	15:14	0x1	Program M domain clock duty cycle
MPLL_X1_CLK_SKEW	18:16	0x0	
MPLL_X2_CLK_SKEW	22:20	0x0	
MPLL_MODE	27:24	0x4	

M domain PLL control

MDLL_CKO - RW - 32 bits - CLKIND:0xF			
Field Name	**Bits**	**Default**	**Description**
MCKOA_SLEEP	0	0x1	0=Enabled 1=PowerDown
MCKOA_RESET	1	0x1	0=Enabled 1=Reset
MCKOA_RANGE	3:2	0x2	
ERSTA_SOUTSEL	5:4	0x1	1=no delay 2=2 elements 3=4 elements
MCKOA_FB_SEL	7:6	0x1	0=raw feedback 1=delayed feedback 2=internal feedback
MCKOA_REF_SKEW	10:8	0x0	
MCKOA_FB_SKEW	14:12	0x0	
MCKOA_BP_SEL	15	0x1	0=DLL clock 1=MCLK

No description available for this register.

MDLL_RDCKA - RW - 32 bits - CLKIND:0x10			
Field Name	**Bits**	**Default**	**Description**
MRDCKA0_SLEEP	0	0x1	0=Enabled 1=PowerDown
MRDCKA0_RESET	1	0x1	0=Enabled 1=Reset
MRDCKA0_RANGE	3:2	0x2	
MRDCKA0_REF_SEL	5:4	0x0	0=MCLK 1=QSA0 pad 2=delayed HCLK1 feedback
MRDCKA0_FB_SEL	7:6	0x0	0=strobe leaf node 1=logic zero 2=internal feedback
MRDCKA0_REF_SKEW	10:8	0x0	
MRDCKA0_SINSEL	11	0x0	0=read strobe per 32 bits 1=read strobe per 8 bits
MRDCKA0_FB_SKEW	14:12	0x0	

MRDCKA0_BP_SEL	15	0x1	0=DLL clock 1=MCLK
MRDCKA1_SLEEP	16	0x1	0=Enabled 1=PowerDown
MRDCKA1_RESET	17	0x1	0=Enabled 1=Reset
MRDCKA1_RANGE	19:18	0x2	
MRDCKA1_REF_SEL	21:20	0x0	0=MCLK 1=QSA4 pad 2=delayed HCLK1 feedback
MRDCKA1_FB_SEL	23:22	0x0	0=strobe leaf node 1=logic zero 2=internal feedback
MRDCKA1_REF_SKEW	26:24	0x0	
MRDCKA1_SINSEL	27	0x0	0=read strobe per 32 bits 1=read strobe per 8 bits
MRDCKA1_FB_SKEW	30:28	0x0	
MRDCKA1_BP_SEL	31	0x1	0=DLL clock 1=MCLK

No description available for this register.

MCLK_CNTL - RW - 32 bits - CLKIND:0x12			
Field Name	**Bits**	**Default**	**Description**
MCLKA_SRC_SEL	2:0	0x0	0=PCICLK 1=MPLLCLK/1 2=MPLLCLK/2 3=MPLLCLK/4 4=MPLLCLK/8 5=External Source/Scan Clock (GPIO13 pin) 6=Test clock from test controller 7=SPLL
YCLKA_SRC_SEL	6:4	0x0	0=PCICLK 1=MPLLCLK/1 2=MPLLCLK/2 3=MPLLCLK/4 4=MPLLCLK/8 5=External Source/Scan Clock (GPIO12 pin) 6=Test clock from test controller 7=Stopped
FORCE_MCLKA	16	0x1	0=Dynamic 1=ForceOn

FORCE_YCLKA	18	0x1	0=Dynamic 1=ForceOn
FORCE_MC	20	0x1	0=Dynamic 1=ForceOn
FORCE_AIC	21	0x1	0=Dynamic 1=ForceOn
MRDCKA0_SOUTSEL	25:24	0x0	0=DLL output clock 1=QS pin 2=QS delayed 2 elements 3=QS delayed 4 elements
MRDCKA1_SOUTSEL	27:26	0x0	0=DLL output clock 1=QS pin 2=QS delayed 2 elements 3=QS delayed 4 elements

No description available for this register.

MCLK_MISC - RW - 32 bits - CLKIND:0x1F			
Field Name	**Bits**	**Default**	**Description**
SCLK_SOURCED_FROM_MPLL_SEL	1:0	0x0	0=MPLLBY1 1=MPLLBY1b 2=MPLLBY2 3=MPLLBY2b
MCLK_FROM_SPLL_DIV_SEL	2	0x0	0=selection of MYCLK_SOURCED_FROM_SPLL_SEL 1=MYCLK_SOURCED_FROM_SPLL_SEL divided BY2
ENABLE_SCLK_FROM_MPLL	3	0x0	0=clock stop 1=clock run
DLL_READY_LAT	8	0x0	0=min latency 1=max latency
CGM_CLK_TO_OUTPIN	16	0x0	0=disabled 1=send out selected clock for jitter test
CLK_OR_COUNT_SEL	17	0x0	0=select TST_MUX_CLK 1=select COUNT_UP
CGM_SPARE	18	0x0	
CGM_SPARE_RD (R)	21:20	0x0	

No description available for this register.

CG_TEST_MACRO_RW_WRITE - RW - 32 bits - CLKIND:0x16			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_WRITE1	13:0	0x0	
TEST_MACRO_RW_WRITE2	27:14	0x0	

M domain PLL control

CG_TEST_MACRO_RW_READ - RW - 32 bits - CLKIND:0x17			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_READ1	15:0	0x0	
TEST_MACRO_RW_READ2	31:16	0x0	

No description available for this register.

CG_TEST_MACRO_RW_DATA - R - 32 bits - CLKIND:0x18			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_DATA	31:0	0x0	

No description available for this register.

CG_TEST_MACRO_RW_CNTL - RW - 32 bits - CLKIND:0x19			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_START	0	0x0	
TEST_MACRO_RW_OP	3:1	0x0	
TEST_MACRO_RW_MODE	5:4	0x0	
TEST_MACRO_RW_MISMATCH_SEL	14:6	0x0	
TEST_MACRO_RW_MISMATCH (R)	15	0x0	
TEST_MACRO_RW_ENABLE	16	0x0	
TEST_MACRO_RW_SCLK_NEG_ENABLE	17	0x0	

No description available for this register.

CLOCK_CNTL_INDEX - RW - 32 bits - [IOReg,MMReg:0x8]			
Field Name	**Bits**	**Default**	**Description**
PLL_ADDR (DISPLAY) (CG) (CGM)	5:0	0x0	Indirect CG and PLL register index
PLL_WR_EN (DISPLAY) (CG) (CGM)	7	0x0	Indirect CG and PLL register write enable 0=Disable writes to CLOCK_CNTL_DATA 1=Enable writing to CLOCK_CNTL_DATA
PPLL_DIV_SEL (DISPLAY) (CG) (CGM)	9:8	0x0	Pixel clock PLL feedback division selection for non-VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=1. See GENMO_WT.VGA_CKSEL for pixel clock selection when VGA mode active. 0=PPLL_DIV0 1=PPLL_DIV1 2=PPLL_DIV2 3=PPLL_DIV3

CG and PLL indirect register control

CLOCK_CNTL_DATA - RW - 32 bits - [IOReg,MMReg:0xC]			
Field Name	**Bits**	**Default**	**Description**
PLL_DATA (DISPLAY) (CG) (CGM)	31:0	0x0	Data for write to CG and PLL indirect registers

Data for write to CG and PLL indirect registers

M_SPLL_REF_FB_DIV - RW - 32 bits - CLKIND:0xA			
Field Name	**Bits**	**Default**	**Description**
M_SPLL_REF_DIV (CG) (CGM)	7:0	0x0	Memory/System PLL reference division
MPLL_FB_DIV (CGM)	15:8	0x0	Memory PLL feedback division
SPLL_FB_DIV (CG)	23:16	0x0	System PLL feedback division

MPLL_REF_SRC_SEL (CGM)	24	0x0	0=Normal 1=Cleaner one through I/O

Memory/System PLL reference/feedback division configuration

PLL_TEST_CNTL - RW - 32 bits - CLKIND:0x13			
Field Name	**Bits**	**Default**	**Description**
TST_SRC_SEL (CG) (CGM)	6:0	0x0	Select different clock source to readable counter and as test clock output
TST_SRC_INV (CG) (CGM)	7	0x0	Invert test clock output
TST_DIVIDERS (CG) (CGM)	8	0x0	Enable driver test
PLL_MASK_READ_B (CG) (CGM)	9	0x1	Mask test clock output and 3 LSBs of readable counter
TESTCLK_MUX_SEL (CG)	12	0x0	0=cg 1=cgm
ANALOG_MON (DISPLAY)	23:15	0x0	
TEST_COUNT (R) (CG) (CGM)	31:24	0x0	Readable counter read back

CG and PLL test control

2.8 clkcntl Registers

CLOCK_CNTL_INDEX - RW - 32 bits - [IOReg,MMReg:0x8]			
Field Name	**Bits**	**Default**	**Description**
PLL_ADDR (DISPLAY) (CG) (CGM)	5:0	0x0	Indirect CG and PLL register index
PLL_WR_EN (DISPLAY) (CG) (CGM)	7	0x0	Indirect CG and PLL register write enable 0=Disable writes to CLOCK_CNTL_DATA 1=Enable writing to CLOCK_CNTL_DATA
PPLL_DIV_SEL (DISPLAY) (CG) (CGM)	9:8	0x0	Pixel clock PLL feedback division selection for non-VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=1. See GENMO_WT.VGA_CKSEL for pixel clock selection when VGA mode active. 0=PPLL_DIV0 1=PPLL_DIV1 2=PPLL_DIV2 3=PPLL_DIV3

CG and PLL indirect register control

CLOCK_CNTL_DATA - RW - 32 bits - [IOReg,MMReg:0xC]			
Field Name	**Bits**	**Default**	**Description**
PLL_DATA (DISPLAY) (CG) (CGM)	31:0	0x0	Data for write to CG and PLL indirect registers

Data for write to CG and PLL indirect registers

PPLL_DIV_0 - RW - 32 bits - CLKIND:0x4			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB0_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST0_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 0

PPLL_DIV_1 - RW - 32 bits - CLKIND:0x5			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB1_DIV (CG)	10:0	0x1f2	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.

PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST1_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 1

PPLL_DIV_2 - RW - 32 bits - CLKIND:0x6			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB2_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST2_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 2

PPLL_DIV_3 - RW - 32 bits - CLKIND:0x7			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB3_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST3_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 3

VCLK_ECP_CNTL - RW - 32 bits - CLKIND:0x8			
Field Name	**Bits**	**Default**	**Description**
VCLK_SRC_SEL	1:0	0x0	Selects source of PIXCLK. The output of this mux selection is post divided by PPLL_POSTx_DIV to create the final PIXCLK. If set to BYTE_CLK, then see BYTE_CLK_POST_DIV below to select the PIXCLK source. Both the clock source you are switching to and from must be running, or the switch will not occur. 0=CPUCLK 1=PSCANCLK 2=BYTE_CLK 3=PPllClk

VCLK_INVERT	4	0x0	Used to invert PPllClk to get opposite duty cycle. Only takes effect when VCLK_SRC_SEL is using PPllClk, and PPLL_POSTx_DIV is divide-by-1. Don't care in other cases. 0=Not Invert 1=Invert
PIXCLK_ALWAYS_ONb	6	0x0	0=PIXCLK is on regardless of CRTC_PIX_WIDTH field (should be set to zero in VGA mode) 1=PIXCLK is off if CRTC_PIX_WIDTH is 0 otherwise PIX-CLK is on
PIXCLK_DAC_ALWAYS_ONb	7	0x0	0=PIXCLK is always on 1=PIXCLK is off during blank time
ECP_DIV	9:8	0x0	OV0CLK clock speed select. Should always be set to 00 for PIXCLK <= 175 MHz and to 01 for PIXCLK > 175 MHz. When set to 01, the overlay and subpic horizontal scale parameters must be adjusted for the extra horizontal replication this causes. Settings 10 and 11 are reserved. 0=VCLK 1=VCLK/2
ECP_FORCE_ON	18	0x0	Controls the dynamic clock control for the back-end over-lay/scaler. Set to low for power reduction. 0=SCALER ACTIVITY 1=CONTINUOUS
SUBCLK_FORCE_ON	19	0x0	0=SUBPIC ACTIVITY 1=CONTINUOUS

General controls for the display clocks.
VCLK is the pixel, or dot, clock.
ECP is the overlay/scaler clock.

PIXCLKS_CNTL - RW - 32 bits - CLKIND:0x2D			
Field Name	**Bits**	**Default**	**Description**
PIX2CLK_SRC_SEL	1:0	0x0	0=PIX2CLK_SRC = CPUCLK (input pin) 1=PIX2CLK_SRC = PSCANCLK (input pin) 2=Reserved 3=PIX2CLK_SRC = P2PllClk
PIX2CLK_INVERT	4	0x0	0=Not Invert 1=Invert
PIX2CLK_ALWAYS_ONb	6	0x0	0=PIX2CLK is on regardless of CRTC2_PIX_WIDTH field 1=PIX2CLK is off if CRTC2_PIX_WIDTH is 0 otherwise PIX2CLK is on
PIX2CLK_DAC_ALWAYS_ONb	7	0x0	0=PIX2CLK is always on 1=PIX2CLK is off during blank time

PIXCLK_TV_SRC_SEL	8	0x0	0=PIXCLK_TV = PIXCLK 1=PIXCLK_TV = PIX2CLK
PIXCLK_BLEND_ALWAYS_ONb	11	0x0	0=1 = Enable dynamic stopping for PIXCLK_BLEND
PIXCLK_GV_ALWAYS_ONb	12	0x0	0=1 = Enable dynamic stopping for PIXCLK_GV
PIXCLK_DIG_TMDS_ALWAYS_ONb	13	0x0	
PIXCLK_LVDS_ALWAYS_ONb	14	0x0	
PIXCLK_TMDS_ALWAYS_ONb	15	0x0	

No description available for this register.

PLL_TEST_CNTL - RW - 32 bits - CLKIND:0x13			
Field Name	**Bits**	**Default**	**Description**
TST_SRC_SEL (CG) (CGM)	6:0	0x0	Select different clock source to readable counter and as test clock output
TST_SRC_INV (CG) (CGM)	7	0x0	Invert test clock output
TST_DIVIDERS (CG) (CGM)	8	0x0	Enable driver test
PLL_MASK_READ_B (CG) (CGM)	9	0x1	Mask test clock output and 3 LSBs of readable counter
TESTCLK_MUX_SEL (CG)	12	0x0	0=cg 1=cgm
ANALOG_MON (DISPLAY)	23:15	0x0	
TEST_COUNT (R) (CG) (CGM)	31:24	0x0	Readable counter read back

CG and PLL test control

P2PLL_DIV_0 - RW - 32 bits - CLKIND:0x2C			
Field Name	**Bits**	**Default**	**Description**
P2PLL_FB_DIV (CG)	10:0	0x0	
P2PLL_ATOMIC_UPDATE_W *(mirror of P2PLL_REF_DIV:P2PLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	
P2PLL_ATOMIC_UPDATE_R *(mirror of P2PLL_REF_DIV:P2PLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	
P2PLL_POST_DIV (DISPLAY)	18:16	0x0	0=V2CLK = V2CLK_SRC 1=V2CLK = V2CLK_SRC/2 2=V2CLK = V2CLK_SRC/4 3=V2CLK = V2CLK_SRC/8 4=V2CLK = V2CLK_SRC/3 5=reserved 6=V2CLK = V2CLK_SRC/6 7=V2CLK = V2CLK_SRC/12

No description available for this register.

2.9 MemoryController Registers

AGP_BASE - RW - 32 bits - [MMReg:0x170]			
Field Name	**Bits**	**Default**	**Description**
AGP_BASE_ADDR	31:0	0x0	When a request falls in the internal AGP aperture (MC_AGP_LOCATION), a relative address is formed by stripping off MC_AGP_START. AGP_BASE_ADDR is added to the relative address to create the address in the host system. NOTE: Bits 0:21 of this field are hardwired to ZERO.

Specifies the base location of AGP space in the host system.

MEM_CNTL - RW - 32 bits - [MMReg:0x140]			
Field Name	**Bits**	**Default**	**Description**
DISABLE_AP	2	0x0	When set, instructs MC to disable precharging of banks. Useful only for HW debug or auto-detection of memory configuration. NOTE: Not supported for Rage 6A. 0=Auto-Precharge Enabled 1=Auto-Precharge Disabled
HALF_MODE	3	0x0	0=64 bit interface support channel A 1=32 bit interface support channel A
MEM_BANK_MAPPING_A	7:4	0x0	Reserved for future use. 0=Reserved

MEM_ADDR_MAPPING_A	15:8	0x0	Defines the memory configuration in the A channel. (Bit definitions relative to 8-bit field) (7) : Ranks - Groups of devices sharing common CS 0 = 1 rank 1 = 2 ranks (6:5) : Banks per device 0 = 2 banks 1 = 4 banks 2 = 8 banks (not supported for Rage 6A) 3 = not supported (4:2) : Rows per bank 0 = 2**9 rows 1 = 2**10 rows 2 = 2**11 rows 3 = 2**12 rows 4 = 2**13 rows 5-7 = not supported (1:0) : Columns per page 0 = 128 columns (not supported for Rage 6A) 1 = 256 columns 2 = 512 columns 3 = not supported

This register defines the physical configruation of frame buffer memory and how it is mapped to the chip internal address space.

EXT_MEM_CNTL - RW - 32 bits - [MMReg:0x144]			
Field Name	**Bits**	**Default**	**Description**
MEM_TRP	1:0	0x0	Min time from Precharge to RAS in same bank 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
MEM_TRCD	3:2	0x0	Min RAS to CAS delay in same bank 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks

MEM_TRAS	6:4	0x0	Min time between RAS operations in same bank 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks 4=5 clocks 5=6 clocks 6=7 clocks 7=8 clocks
MEM_TRRD	9:8	0x0	Min time between RAS operations in different banks of same device 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
MEM_TR2W	11:10	0x0	Min Read to Write turnaround time 0=0 clock 1=1 clocks 2=2 clocks 3=3 clocks
MEM_TWR	13:12	0x0	Write Recovery time 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
MEM_TW2R	15:14	0x0	Write to Read turnaround time Not implemented in Rage 6A - see MEM_TWR_MODE 0=0 clock 1=1 clocks 2=2 clocks 3=3 clocks
MEM_TR2R	17:16	0x0	Read to Read (rank switch) time 0=0 clock 1=1 clocks 2=2 clocks 3=3 clocks
MEM_TWR_MODE	18	0x0	When set, use TWR timer for TW2R. Otherwise, use 0 clocks. 0=Ignore TWR Rule 1=Use TWR Rule
MEM_REFRESH_DIS	20	0x0	Disables all refreshing when set 0=Not Disabled 1=Disabled
MEM_REFRESH_RATE	31:24	0x0	One memory refresh is performed every (MEM_REFRESH_RATE+1) * 64 MCLK cycles.

Defines the SDRAM timing parameters and refresh rate.

MC_FB_LOCATION - RW - 32 bits - [MMReg:0x148]			
Field Name	**Bits**	**Default**	**Description**
MC_FB_START (MC) (HDP)	15:0	0x0	Start of local frame buffer section of 32 bit internal address space. Recommend setting this register to same as CONFIG_APER_0_BASE.APER_0_BASE shr 16. NOTE: Bits 0:5 of this field are hardwired to ZERO.
MC_FB_TOP (MC)	31:16	0x3f	End of local frame buffer section of 32 bit internal address space. Recommand setting this register to (CONFIG_APER_0_BASE.APER_0_BASE + CONFIG_APER_SIZE.APER_SIZE - 1) shr 16. NOTE: Bits 0:5 of this field are hardwired to ONE.

This register defines the location of the frame buffer in the internal address space. The internal address space has 32 address bits. Minimum Frame buffer size for Rage 5 is 2 MB, and the start location is required to be on a 4 MB boundry. Therefore START(21:0) must be 0x000000 and TOP(21:0) must be 0x3FFFFF. Only the 16 MSBs of each are loaded in the register.

MC_AGP_LOCATION - RW - 32 bits - [MMReg:0x14C]			
Field Name	**Bits**	**Default**	**Description**
MC_AGP_START	15:0	0x0	Start location of AGP aperture. Load 16 MSBs only. LSBs assumed to be 0x0. NOTE: Bits 0:5 of this field are hardwired to ZERO.
MC_AGP_TOP	31:16	0x3f	Top location within AGP aperture. Load 16 MSBs only. LSBs assumed to be 0xFFFF. NOTE: Bits 0:5 of this field are hardwired to ONE.

Defines the location of AGP in the chip internal address space.
Aperture must be aligned to a 4 MB boundary.

MEM_INIT_LATENCY_TIMER - RW - 32 bits - [MMReg:0x154]			
Field Name	**Bits**	**Default**	**Description**
MEM_RB0R_INIT_LAT	3:0	0x0	The arbiter will increase the priority of a request (ignoring efficiency) if a request is not selected withing (INIT_LAT*16) MCLKs. Timer is started when request reaches the arbiter, not when request is received by MC. Timeout setting for RB0 read requests

MEM_RB1R_INIT_LAT	7:4	0x0	Timeout setting for RB1 read requests
MEM_PPR_INIT_LAT	11:8	0x0	Timeout setting for Pixel Pipe read requests
MEM_DISPR_INIT_LAT	15:12	0x0	Timeout setting for Display read requests
MEM_RB0W_INIT_LAT	19:16	0x0	Timeout setting for RB0 write requests
MEM_RB1W_INIT_LAT	23:20	0x0	Timeout setting for RB1 write requests
MEM_FIXED_INIT_LAT	27:24	0x0	Timeout setting for fixed priority client arbitration winner. Timer is started when request wins fixed priority arbitration.
SAME_PAGE_PRIO	30:28	0x0	Indicates that (VALUE+1) consecutive page hits are needed before the arbiter may switch to another bank in the same device with no timing penalty.

This register provides controls used to trade off memory efficiency and request latency.

MEM_SDRAM_MODE_REG - RW - 32 bits - [MMReg:0x158]			
Field Name	**Bits**	**Default**	**Description**
MEM_MODE_REG	14:0	0x0	Value to be loaded in SDRAM mode or extended mode register
MEM_RBS_POSITION_A	17:16	0x0	Used to advance or delay capture of read data coming from A channel IO pads 0=Nominal 1=Advance 1 clock 2=Delay 1 clock 3=Not Used
MEM_CAS_LATENCY	22:20	0x0	Must match CAS latency programmed into MEM_MODE_REG 1=1 clock 2=2 clocks 3=3 clocks 5=1.5 clocks 6=2.5 clocks
WR_LAT_EQ_CAS_LAT	23	0x0	Reserved. Must be programmed to '0' 0=Default WR Latency 1=Use CAS Latency for WR Latency
MEM_ERST_POSITION_A	25:24	0x0	Used to advance or delay the A channel ERST signal, which indicates to the IO pads when to begin capturing a burst of read data. 0=Nominal 1=Advance 1 clock 2=Delay 1 clock 3=Not Used

MC_INIT_COMPLETE	28	0x0	As long as this bit is '0', the MC will not accept requests from the clients. Used primarily to block requests when the MC might mis-handle them, such as when the FB or AGP apertures are undefined or unstable. 0=Register Initialization Not Complete 1=Register Initialization Complete
MEM_NON_JEDEC	29	0x0	This bit should be set when using a DDR memory which has no internal delay-locked loop (DLL). Ignored for SDR memory. 0=JEDEC 1=1 Non-JEDEC
MEM_CFG_TYPE	30	0x0	0=SDR 1=DDR
MEM_SDRAM_RESET	31	0x0	The MC will program the SDRAM mode register on a transition from 0 to 1 0=Normal 1=Reset

Register used to control the programming of the SDRAM mode register.
Also contains controls used to compensate for different CAS latency settings, SDR/DDR differences, and variations in board-level timing.

MEM_IO_CNTL_A0 - RW - 32 bits - [MMReg:0x178]			
Field Name	**Bits**	**Default**	**Description**
MEM_N_CKA0	3:0	0x0	N Drive Strength for CLKA0 0x0 = weakest, 0xF = strongest
MEM_N_CKA1	7:4	0x0	N Drive Strength for CLKA1 and CLKAFB 0x0 = weakest, 0xF = strongest
MEM_N_AA	11:8	0x0	N Drive Strength for AA bus, RASAb, CASAb, WEAb, CSAb, and CKEA 0x0 = weakest, 0xF = strongest
MEM_N_DQMA	15:12	0x0	N Drive Strength for DQA and DQMA busses 0x0 = weakest, 0xF = strongest
MEM_N_DQSA	19:16	0x0	N Drive Strength for QSA (strobes) 0x0 = weakest, 0xF = strongest
MC_IO_SSTL_ENA	20	0x0	When set, disables all internal pullup/pulldowns in pads for A channel 0=LVTTL Interface 1=SSTL Interface
MEM_SLEW_CKA0	24	0x0	Slew rate control for CLKA0 0=Low Slew 1=High Slew

MEM_SLEW_CKA1	25	0x0	Slew rate control for CLKA1 and CLKAFB 0=Low Slew 1=High Slew
MEM_SLEW_AA	26	0x0	Slew rate control for AA bus, RASAb, CASAb, WEAb, CSAb, and CKEA 0=Low Slew 1=High Slew
MEM_SLEW_DQMA	27	0x0	Slew rate control for DQA and DQMA busses 0=Low Slew 1=High Slew
MEM_SLEW_DQSA	28	0x0	Slew rate control for QSA (strobes) 0=Low Slew 1=High Slew
MEM_PREAMP_DAA	29	0x0	Enables preamp to compensate for long runs of '0' or '1' on AA bus, RASAb, CASAb, WEAb, CSAb, and CKEA 0=Not Enabled 1=Enabled
MEM_PREAMP_DQMA	30	0x0	Enables preamp to compensate for long runs of '0' or '1' on DQA and DQMA busses 0=Not Enabled 1=Enabled
MEM_PREAMP_DQSA	31	0x0	Enables preamp to compensate for long runs of '0' or '1' on QSA (strobes) 0=Not Enabled 1=Enabled

IO Pad controls for memory A channel

MEM_IO_CNTL_A1 - RW - 32 bits - [MMReg:0x17C]			
Field Name	**Bits**	**Default**	**Description**
MEM_P_CKA0	3:0	0x0	P Drive Strength for CLKA0 0x0 = weakest, 0xF = strongest
MEM_P_CKA1	7:4	0x0	P Drive Strength for CLKA1 and CLKAFB 0x0 = weakest, 0xF = strongest
MEM_P_AA	11:8	0x0	P Drive Strength for AA bus, RASAb, CASAb, WEAb, CSAb, and CKEA 0x0 = weakest, 0xF = strongest
MEM_P_DQMA	15:12	0x0	P Drive Strength for DQA and DQMA busses 0x0 = weakest, 0xF = strongest
MEM_P_DQSA	19:16	0x0	P Drive Strength for QSA (strobes) 0x0 = weakest, 0xF = strongest
DLL_FB_SLCT_CKA	21:20	0x0	Variable delay select for DLL internal feedback path. Normally not used.

CLKA0_ENABLE	22	0x1	Output enable for CLKA0. Always enable during normal operation. 0=Not Enabled 1=Enabled
CLKA1_ENABLE	23	0x1	Output enable for CLKA1. May be disabled only if CLKA1 not used in board layout. 0=Not Enabled 1=Enabled
CLKAFB_ENABLE	24	0x1	Output enable for CLKAFB. Enable for SDR, disable for DDR. 0=Not Enabled 1=Enabled
DFR_DQSA	25	0x0	Selects receiver type for data strobes. Normally use 0 for SDR, 1 for DDR. 0=Schmitt Trigger 1=Differential Rcvr
DFR_CKA	26	0x0	Selects receiver type for clock. Normally use 0 for SDR, 1 for DDR. 0=Schmitt Trigger 1=Differential Rcvr
DFR_DQMA	27	0x0	Selects receiver type for data bus. Normally use 0 for SDR, 1 for DDR. 0=Schmitt Trigger 1=Differential Rcvr
DQS_DRIVER_SLCT_A0	28	0x0	Ouput enable for QSA0. Use 0 for SDR, 1 for DDR. 0=Not On 1=On
DQS_DRIVER_SLCT_A1	29	0x0	Ouput enable for QSA1. Use 0 for SDR and DDR with 1 strobe per 32 bits, 1 for DDR with 1 strobe per 8 bits. 0=Not On 1=On
DQS_DRIVER_SLCT_A2	30	0x0	Ouput enable for QSA2. Use 0 for SDR and DDR with 1 strobe per 32 bits, 1 for DDR with 1 strobe per 8 bits. 0=Not On 1=On
DQS_DRIVER_SLCT_A3	31	0x0	Ouput enable for QSA3. Use 0 for SDR and DDR with 1 strobe per 32 bits, 1 for DDR with 1 strobe per 8 bits. 0=Not On 1=On

IO Pad controls for memory A channel

MEM_IO_OE_CNTL - RW - 32 bits - [MMReg:0x18C]			
Field Name	**Bits**	**Default**	**Description**
MEM_DQ_OE_EXTEND_A	1:0	0x0	Extend the output enable for A channel write data beyond the base of 2 clocks by the selected amount. 0=0.5 clocks 1=1 clock 2=1.5 clocks 3=2 clocks
MEM_DQ_OE_POSITION_A	3:2	0x0	Shift the position of the output enable for A channel write data. 0=Nominal 1=Advance 1 clock 2=Delay 1 clock 3=Not used
MEM_QS_OE_EXTEND_A	5:4	0x0	Extend the output enable for A channel write data strobe beyond the base of 2 clocks by the selected amount. 0=0.5 clocks 1=1 clock 2=1.5 clocks 3=2 clocks
MEM_QS_OE_POSITION_A	7:6	0x0	Shift the position of the output enable for A channel write data strobe. 0=Nominal 1=Advance 1 clock 2=Delay 1 clock 3=Not used
MEM_DYNAMIC_CKE	16	0x0	Allows use of dynamic memory clock enable for power savings 0=Disable 1=Enable
MEM_SDRAM_TRI_EN	17	0x0	When set, the MC will hold all RAS, CAS, CS, and WE pins in hi-Z state during power-up and power mode transitions. 0=Not Allowed 1=Allowed

This register provides timing adjustments for the OEs of bidirectional pins as well as some power management controls.

MC_DEBUG - RW - 32 bits - [MMReg:0x188]			
Field Name	**Bits**	**Default**	**Description**
IGNORE_RW_PENALTY_RB0R	0	0x0	0=Don't Ignore 1=Ignore
IGNORE_RW_PENALTY_RB1R	1	0x0	0=Don't Ignore 1=Ignore

IGNORE_RW_PENALTY_RB0W	2	0x0	0=Don't Ignore 1=Ignore
IGNORE_RW_PENALTY_RB1W	3	0x0	0=Don't Ignore 1=Ignore
IGNORE_RW_PENALTY_DISPR	4	0x0	0=Don't Ignore 1=Ignore
IGNORE_RW_PENALTY_PPR	5	0x0	0=Don't Ignore 1=Ignore
IGNORE_RW_PENALTY_FIXED	6	0x0	0=Don't Ignore 1=Ignore
MEM_VIPW_PRIORITY	9:8	0x0	0=VIPw has lowest priority of RT urgent clients 1=VIPw priority just below SUBP when urgent 2=VIPw priority just below DISP when urgent 3=VIPw has highest priority when urgent
CLKA0b_ENABLE	10	0x0	0=Disable 1=Enable
CLKA1b_ENABLE	11	0x0	0=Disable 1=Enable
CLKB0b_ENABLE	12	0x0	0=Disable 1=Enable
CLKB1b_ENABLE	13	0x0	0=Disable 1=Enable
DEBUG4	14	0x0	0=Disable 1=Enable
MC_BIST_EN	15	0x0	0=Disable 1=Enable
MC_BIST_CTRL	19:16	0x0	0=BIST control bits
DEBUG0	20	0x0	0=Disable 1=Enable
DEBUG1	21	0x0	0=Disable 1=Enable
DEBUG2	22	0x0	0=Disable 1=Enable
DEBUG3	23	0x0	0=Disable 1=Enable
MC_DEBUG	27:24	0x0	
PWR_DOWN_MEM	28	0x0	0=Normal operation 1=Power down memory
SWAP_CS	29	0x0	0=Straight connection for chip selects 1=Swapped connection for chip selects

BLOCK_CS0	30	0x0	0=CS0 enabled 1=CS0 disabled
IKOSE	31	0x0	0=Normal refresh rate 1=Slow refresh rate

No description available for this register.

MC_STATUS - R - 32 bits - [MMReg:0x150]			
Field Name	**Bits**	**Default**	**Description**
MEM_PWRUP_COMPL_A	0	0x0	Indicates that the last SDRAM_RESET operation has completed for the A channel. Do not inititiate a new SDRAM_RESET operation until 'Ready' is indicated. 0=SDRAM Init in Process 1=Ready
MC_IDLE	2	0x0	Indicates that there are no pending or in-process frame buffer requests. Does not include status on pending or in-process requests to system memory. 0=Not Idle 1=Idle
SPARE	15:3	0x0	Reserved

Read-only status register

2.10 RegisterBackboneManager Registers

RBBM_CNTL - RW - 32 bits - [IOReg,MMReg:0xEC] [MMReg:0xE44]			
Field Name	**Bits**	**Default**	**Description**
RB_SETTLE	3:0	0xf	Read Daisy Chain Bus Settling Time. (in clocks + 2; minimum time is 2 clocks) Default = 15
ABORTCLKS_HI	6:4	0x0	The number of clocks that we will allow a non-queued write from the Host Interface to stall before it is aborted. A value of 0 means never abort. Default = 0.
ABORTCLKS_CP	10:8	0x0	The number of clocks that we will allow a non-queued write from the Command Processor to stall before it is aborted. A value of 0 means never abort. Default = 0.
ABORTCLKS_CFIFO	14:12	0x0	The number of clocks that we will allow a write from the Command FIFO to stall before it is aborted. A value of 0 means never abort. Default = 0.
CPQ_DATA_SWAP	17	0x0	Endian Swap Control for writes to the Command Stream Queue. 0 = No swap 1 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA Default = 0
NO_ABORT_IDCT	21	0x0	Any write transaction to the IDCT block cannot be aborted. Default = 0.
NO_ABORT_BIOS	22	0x0	Any write transaction to the BIOS block cannot be aborted. Default = 0.
NO_ABORT_FB	23	0x0	Any write transaction to the Feature Block cannot be aborted. Default = 0.
NO_ABORT_CP	24	0x0	Any write transaction to the Command Processor block cannot be aborted. Default = 0.
NO_ABORT_HI	25	0x0	Any write transaction to the Host Interface block cannot be aborted. Default = 0.
NO_ABORT_HDP	26	0x0	Any write transaction to the Host Data Processor block cannot be aborted. Default = 0.

NO_ABORT_MC	27	0x0	Any write transaction to the Memory Controller block cannot be aborted. Default = 0.
NO_ABORT_AIC	28	0x0	Any write transaction to the AGP Interface block cannot be aborted. Default = 0.
NO_ABORT_VIP	29	0x0	Any write transaction to the VIP block cannot be aborted. Default = 0.
NO_ABORT_DISP	30	0x0	Any write transaction to the Display block cannot be aborted. Default = 0.
NO_ABORT_CG	31	0x0	Any write transaction to the CG or CGM blocks cannot be aborted. Default = 0.

Control Register

RBBM_SOFT_RESET - RW - 32 bits - [IOReg,MMReg:0xF0] [MMReg:0xE48]			
Field Name	**Bits**	**Default**	**Description**
SOFT_RESET_CP	0	0x0	Soft Reset to the Command Processor block. Default = 0
SOFT_RESET_HI	1	0x0	Soft Reset to the Host Interface block. Default = 0
SOFT_RESET_SE	2	0x0	Soft Reset to the 3D Setup Engine block. Default = 0
SOFT_RESET_RE	3	0x0	Soft Reset to the 3D Raster Engine block. Default = 0
SOFT_RESET_PP	4	0x0	Soft Reset to the 3D Pixel Pipe block. Default = 0
SOFT_RESET_E2	5	0x0	Soft Reset to the 2D Engine block. Default = 0
SOFT_RESET_RB	6	0x0	Soft Reset to the 2D/3D Render Backend block. Default = 0
SOFT_RESET_HDP	7	0x0	Soft Reset to the Host Data Processor block. Default = 0
SOFT_RESET_MC	8	0x0	Soft Reset to the Memory Controller block. Default = 0
SOFT_RESET_AIC	9	0x0	Soft Reset to the AGP Interface block. Default = 0
SOFT_RESET_VIP	10	0x0	Soft Reset to the VIP block. Default = 0

SOFT_RESET_DISP	11	0x0	Soft Reset to the Display block. Default = 0
SOFT_RESET_CG	12	0x0	Soft Reset to Clock Generator blocks. Default = 0

Soft Reset Generation

RBBM_STATUS - R - 32 bits - [MMReg:0xE40] [MMReg:0x1740]			
Field Name	**Bits**	**Default**	**Description**
CMDFIFO_AVAIL	6:0	0x40	Number of available entries (doublewords) in the Command FIFO. Default = 64
HIRQ_ON_RBB	8	0x0	There is a request from the Host Interface on the backbone.
CPRQ_ON_RBB	9	0x0	There is a request from the Command Processor on the backbone.
CFRQ_ON_RBB	10	0x0	There is a request from the Command FIFO on the back-bone.
HIRQ_IN_RTBUF	11	0x0	There is a request from the Host Interface present in the Retry Buffer.
CPRQ_IN_RTBUF	12	0x0	There is a request from the Command Processor present in the Retry Buffer.
CFRQ_IN_RTBUF	13	0x0	There is a request from the Command FIFO present in the Retry Buffer.
CF_PIPE_BUSY	14	0x0	The Command FIFO pipeline is busy. This indicates that there is something in the command FIFO, or a command fifo request is present further down in the RBBM pipe, eg, on the backbone.
ENG_EV_BUSY	15	0x0	The RBBM's Event Engine is Busy. That is, we are waiting for a WAIT_UNTIL synchronization event.
CP_CMDSTRM_BUSY	16	0x0	The Command Processor's Command Stream is Busy. This covers from the Ring Buffer (in memory) through the CP's internal processing pipeline.
E2_BUSY	17	0x0	2D Engine is Busy
RB2D_BUSY	18	0x0	2D portion of Render Backend is Busy
RB3D_BUSY	19	0x0	3D portion of Render Backend is Busy
SE_BUSY	20	0x0	3D Setup Engine is Busy
RE_BUSY	21	0x0	3D Raster Engine is Busy

TAM_BUSY	22	0x0	3D Texture Address Module (sub-block of Pixel Pipe) is Busy
TDM_BUSY	23	0x0	3D Texture Data Module (sub-block of Pixel Pipe) is Busy
PB_BUSY	24	0x0	3D Pixel Blender (sub-block of Pixel Pipe) is Busy
GUI_ACTIVE	31	0x0	2D engine busy OR 3D engine busy OR Command FIFO not empty OR CP Microengine busy OR Command Stream Queue not empty OR Ring Buffer not empty

Status Register

ISYNC_CNTL - RW - 32 bits - [MMReg:0x1724]			
Field Name	**Bits**	**Default**	**Description**
ISYNC_ANY2D_IDLE3D	0	0x0	A write to any 2D Engine register stalls if the 3D Engine is busy OR the Render Backend is not clean of 3D destination data OR the Memory Controller is not clean of 3D destination data. Default = 0.
ISYNC_ANY3D_IDLE2D	1	0x0	A write to any 3D Engine register stalls if the 2D Engine is busy OR the Render Backend is not clean of 3D destination data OR the Memory Controller is not clean of 3D destination data. Default = 0.
ISYNC_TRIG2D_IDLE3D	2	0x0	A write to any 2D trigger register stalls if the 3D Engine is busy OR the Render Backend is not clean of 3D destination data OR the Memory Controller is not clean of 3D destination data. The 2D trigger registers are: DST_WIDTH_HEIGHT DST_HEIGHT_WIDTH DST_HEIGHT_WIDTH_8 DST_WIDTH_X DST_WIDTH_X_INCY DP_GUI_MASTER_CNTL DP_DATATYPE DP_WRITE_MSK DP_MIX CLR_CMP_CNTL RB2D_WRITEBACK_DATA_LO RB2D_WRITEBACK_DATA_HI RB2D_WRITEBACK_ADDR RB2D_DSTCACHE_CTLSTAT Default = 0.

ISYNC_TRIG3D_IDLE2D	3	0x0	A write to any 3D trigger register stalls if the 2D Engine is busy OR the Render Backend is not clean of 2D destination data OR the Memory Controller is not clean of 2D destination data. The 3D trigger registers are: SE_VF_CNTL RE_E2_3 PP_CNTL PP_MC_CONTEXT RB3D_WRITEBACK_DATA_LO RB3D_WRITEBACK_DATA_HI RB3D_WRITEBACK_ADDR RB3D_CNTL RB3D_PLANEMASK RB3D_BLENDCNTL RB3D_ROPCNTL RB3D_DSTCACHE_CTLSTAT Default = 0.
ISYNC_WAIT_IDLEGUI	4	0x0	A write to the WAIT_UNTIL register stalls if the 2D or 3D Engine is busy. Default = 0.
ISYNC_CPSCRATCH_IDLEGUI	5	0x0	A write to any of the CP's GUI ScratchPad Registers stalls if the 2D or 3D Engine is busy. (For Rage128 compatibility) Default = 0.

Implicit Synchronization Control

RBBM_GUICNTL - RW - 32 bits - [MMReg:0x172C]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA_SWAP	1:0	0x0	Endian Swap Control for 'Host Data' writes to 2D engine. 0 = No swap 1 = 16-bit swap: 0xAABBCCDD becomes 0xBBAADDCC 2 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 3 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB Default = 0

GUI Control

RBBM_CMDFIFO_ADDR - W - 32 bits - [MMReg:0xE70]			
Field Name	**Bits**	**Default**	**Description**
CMDFIFO_ADDR	5:0	0x0	Address into the Command FIFO which is to be read from. Used for debug, to read the contents of the Command FIFO.

Command FIFO Address

RBBM_CMDFIFO_DATAL - R - 32 bits - [MMReg:0xE74]			
Field Name	**Bits**	**Default**	**Description**
CMDFIFO_DATAL	31:0	0x0	Data from the Command FIFO, from location pointed to by the RBBM_CMDFIFO_ADDR register. Used for debug, to read the contents of the Command FIFO.

Command FIFO Data Low

RBBM_CMDFIFO_DATAH - R - 32 bits - [MMReg:0xE78]			
Field Name	**Bits**	**Default**	**Description**
CMDFIFO_DATAH	11:0	0x0	Upper Data from the Command FIFO, from location pointed to by the RBBM_CMDFIFO_ADDR register. Used for debug, to read the contents of the Command FIFO. These bits contain the register-space Address and the Byte Enables.

Command FIFO Data High

RBBM_CMDFIFO_STAT - R - 32 bits - [MMReg:0xE7C]			
Field Name	**Bits**	**Default**	**Description**
CMDFIFO_RPTR	5:0	0x0	Current Read Pointer into the Command FIFO. Default = 0.
CMDFIFO_WPTR	13:8	0x0	Current Write Pointer into the Command FIFO. Default = 0.

Command FIFO Status

WAIT_UNTIL - RW - 32 bits - [MMReg:0x1720]			
Field Name	**Bits**	**Default**	**Description**
WAIT_CRTC_PFLIP	0	0x0	Wait for the 'Pending Flip' signal to be OFF
WAIT_RE_CRTC_VLINE	1	0x0	Wait for Rising Edge of CRTC_VLINE signal
WAIT_FE_CRTC_VLINE	2	0x0	Wait for Falling Edge of CRTC_VLINE signal
WAIT_CRTC_VLINE	3	0x0	Wait for CRTC_VLINE signal to be ON

WAIT_DMA_VIPH0_IDLE	4	0x0	Wait for VIP Host DMA Channel 0 to be idle
WAIT_DMA_VIPH1_IDLE	5	0x0	Wait for VIP Host DMA Channel 1 to be idle
WAIT_DMA_VIPH2_IDLE	6	0x0	Wait for VIP Host DMA Channel 2 to be idle
WAIT_DMA_VIPH3_IDLE	7	0x0	Wait for VIP Host DMA Channel 3 to be idle
WAIT_DMA_VID_IDLE	8	0x0	Wait for VID DMA Channel to be idle
WAIT_DMA_GUI_IDLE	9	0x0	Wait for GUI DMA Channel to be idle
WAIT_CMDFIFO	10	0x0	Wait until there are at least CMDFIFO_ENTRIES number of occupied entries in the Command FIFO. Used to 'batch-up' a group of writes, to ensure they get written within a short time period of each other.
WAIT_OV0_FLIP	11	0x0	Wait for Overlay Flip signal to be ON
WAIT_OV0_SLICEDONE	12	0x0	Wait for Overlay Slice Done signal to be ON
WAIT_2D_IDLE	14	0x0	Wait for 2D engine (including 2D render backend) to be idle
WAIT_3D_IDLE	15	0x0	Wait for 3D engine (including 3D render backend) to be idle
WAIT_2D_IDLECLEAN	16	0x0	Wait for 2D engine to be idle and 2D render backend to be clean and Memory Controller to be clean of 2D destination data
WAIT_3D_IDLECLEAN	17	0x0	Wait for 3D engine to be idle and 3D render backend to be clean and Memory Controller to be clean of 3D destination data
WAIT_HOST_IDLECLEAN	18	0x0	Wait for Host Interface/Host Data Path to be idle and clean
WAIT_EXTERN_SIG	19	0x0	Wait for External Signal (from VIP block) to be ON
CMDFIFO_ENTRIES	26:20	0x0	Number of entries to wait for if the WAIT_CMDFIFO bit is ON
WAIT_BOTH_CRTC_PFLIP	30	0x0	Wait for both CRTC1_PFLIP and CRTC2_PFLIP
ENG_DISPLAY_SELECT	31	0x0	Selects VLINE and PFLIP to be used in 0-3 wait conditions

This register is for Explicit Synchronization.
A write to this register will travel through the Command FIFO, and when it gets to the bottom of the FIFO, the write will stall until its condition(s) is/are met. If multiple conditions are being tested, they *all* must be true in order to un-stall the write. While it is stalled, other writes can continue to fill-up the Command FIFO behind it.

NQWAIT_UNTIL - W - 32 bits - [MMReg:0xE50]			
Field Name	**Bits**	**Default**	**Description**
WAIT_GUI_IDLE	0	0x0	Wait for 2D and 3D engines to be idle, and for Command FIFO to be empty.

This register is for Explicit Synchronization.
A write to this register will stall *at the top of the Command FIFO*, so that the entire FIFO has to flush before this write can become un-stalled. This technique helps you control ordering of a write to a non-queued register (particularly, the IDCT registers, which are the only non-queued registers in the Command Processor's address-space), to make sure it happens after a certain group of writes to the queued registers.
This register is not implemented in the host's address space; only the CP's

RBBM_DEBUG - RW - 32 bits - [MMReg:0xE6C]			
Field Name	**Bits**	**Default**	**Description**
RBBM_DEBUG	31:0	0x0	Reserved bits

This register is for Explicit Synchronization.
A write to this register will travel through the Command FIFO, and when it gets to the bottom of the FIFO, the write will stall until its condition(s) is/are met. If multiple conditions are being tested, they *all* must be true in order to un-stall the write. While it is stalled, other writes can continue to fill-up the Command FIFO behind it.

GEN_INT_CNTL - RW - 32 bits - [IOReg,MMReg:0x40]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_MASK (DISPLAY)	0	0x0	Vertical blank interrupt mask. 0=Disable 1=Enable
CRTC_VLINE_MASK (DISPLAY)	1	0x0	Vertical line interrupt mask. 0=Disable 1=Enable
CRTC_VSYNC_MASK (DISPLAY)	2	0x0	Vertical sync interrupt mask. 0=Disable 1=Enable
SNAPSHOT_MASK (DISPLAY)	3	0x0	Snapshot interrupt mask. 0=Disable 1=Enable
FP_DETECT_MASK (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt mask. 0=Disable 1=Enable

CRTC2_VLINE_MASK (DISPLAY)	5	0x0	0=Disable 1=Enable
CRTC2_VSYNC_MASK (DISPLAY)	6	0x0	0=Disable 1=Enable
SNAPSHOT2_MASK (DISPLAY)	7	0x0	0=Disable 1=Enable
CRTC2_VBLANK_MASK (DISPLAY)	9	0x0	0=Disable 1=Enable
FP2_DETECT_MASK (DISPLAY)	10	0x0	0=Disable 1=Enable
VSYNC_DIFF_OVER_LIMIT_MASK (TVOUT)	11	0x0	0=Disable 1=Enable
GUI_IDLE_MASK (RBBM)	19	0x0	GUI idle interrupt mask. 0=Disable 1=Enable
SW_INT_EN (HDP)	25	0x0	Software interrupt mask. 0=Disable 1=Enable
GEYSERVILLE_MASK (VIP)	27	0x0	0=Disable 1=Enable
HDCP_AUTHORIZED_INT_MASK (DISPLAY)	28	0x0	0=Disable 1=Enable
DVI_I2C_INT_MASK (DISPLAY)	29	0x0	0=Disable 1=Enable
GUIDMA_MASK (CP)	30	0x0	GUI DMA channel interrupt mask. 0=Disable 1=Enable
reserved	31		reserved

General Interrupt Control register.
The MASK/EN fields control whether the respective status bits are enabled to drive the system interrupt pin. Even if enabled here, the interrupt line is not driven unless enabled in the PCI configuration space.

GEN_INT_STATUS - RW - 32 bits - [IOReg,MMReg:0x44]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_STAT (R) (DISPLAY)	0	0x0	Vertical blank interrupt. Set when display in vertical retrace. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VBLANK_STAT_AK (W) (DISPLAY)	0	0x0	Vertical blank interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VLINE_STAT (R) (DISPLAY)	1	0x0	Vertical line interrupt. Set on display line on programmed by the CRTC_VLINE_CRNT_VLINE.CRTC_VLINE register. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VLINE_STAT_AK (W) (DISPLAY)	1	0x0	Vertical line interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VSYNC_STAT (R) (DISPLAY)	2	0x0	Vertical sync interrupt. Set on start of VSYNC at the DAC. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_STAT_AK (W) (DISPLAY)	2	0x0	Vertical sync interrupt acknowledge. 0=No effect 1=Clear status
SNAPSHOT_STAT (R) (DISPLAY)	3	0x0	Snapshot interrupt. Set as controlled by SNAPSHOT_VIF_COUNT register. 0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT_STAT_AK (W) (DISPLAY)	3	0x0	Snapshot interrupt acknowledge. 0=No effect 1=Clear status
FP_DETECT_STAT (R) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt. Set on HPD connect or disconnect as controlled by FP_GEN_CNTL.FP_DETECT_INT_POL. 0=No event 1=Event has occurred, interrupting if enabled
FP_DETECT_STAT_AK (W) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt acknowledge. 0=No effect 1=Clear status
CRTC2_VLINE_STAT (R) (DISPLAY)	5	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VLINE_STAT_AK (W) (DISPLAY)	5	0x0	0=No effect 1=Clear status

CRTC2_VSYNC_STAT (R) (DISPLAY)	6	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VSYNC_STAT_AK (W) (DISPLAY)	6	0x0	0=No effect 1=Clear status
SNAPSHOT2_STAT (R) (DISPLAY)	7	0x0	0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT2_STAT_AK (W) (DISPLAY)	7	0x0	0=No effect 1=Clear status
CAP0_INT_ACTIVE (R) (VIP)	8	0x0	Capture port 0 has active interrupt(s). 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CRTC2_VBLANK_STAT (R) (DISPLAY)	9	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VBLANK_STAT_AK (W) (DISPLAY)	9	0x0	0=No effect 1=Clear status
FP2_DETECT_STAT (R) (DISPLAY)	10	0x0	0=No event 1=Event has occurred, interrupting if enabled
FP2_DETECT_STAT_AK (W) (DISPLAY)	10	0x0	0=No effect 1=Clear status
VSYNC_DIFF_OVER_LIMIT_STAT (R) (TVOUT)	11	0x0	0=No event 1=Event has occurred, interrupting if enabled
VSYNC_DIFF_OVER_LIMIT_STAT_AK (W) (TVOUT)	11	0x0	0=No effect 1=Clear status
GUI_IDLE_STAT (R) (RBBM)	19	0x1	GUI idle interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_STAT_AK (W) (RBBM)	19	0x0	GUI idle interrupt acknowledge. 0=No effect 1=Clear status
SW_INT (R) (HDP)	25	0x0	Software interrupt. General purpose interrupt that can only be set by software event by writing to SW_INT_SET. 0=No event 1=Event has occurred, interrupting if enabled

SW_INT_AK (W) (HDP)	25	0x0	Software interrupt acknowledge. 0=No effect 1=Clear SW_INT (set low)
SW_INT_SET (W) (HDP)	26	0x0	Software interrupt trigger. 0=No effect 1=Set SW_INT active (high)
GEYSERVILLE_STAT (R) (VIP)	27	0x0	0=No event 1=Event has occurred, interrupting if enabled
GEYSERVILLE_STAT_AK (W) (VIP)	27	0x0	0=No effect 1=Clear status
HDCP_AUTHORIZED_INT_STAT (R) (DISPLAY)	28	0x0	0=No event 1=Event has occurred, interrupting if enabled
HDCP_AUTHORIZED_INT_AK (W) (DISPLAY)	28	0x0	0=No effect 1=Clear status
DVI_I2C_INT_STAT (R) (DISPLAY)	29	0x0	0=No event 1=Event has occurred, interrupting if enabled
DVI_I2C_INT_AK (W) (DISPLAY)	29	0x0	0=No effect 1=Clear status
GUIDMA_STAT (R) (CP)	30	0x0	GUI DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUIDMA_AK (W) (CP)	30	0x0	GUI DMA channel interrupt acknowledge. 0=No effect 1=Clear status
VIDDMA_STAT (R) (CP)	31	0x0	Video capture DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
VIDDMA_AK (W) (CP)	31	0x0	Video capture DMA channel interrupt acknowledge. 0=No effect 1=Clear status

General Interrupt Status register.
These fields can be polled and acknowledged even if interrupts are disabled, or the respective fields are masked in the GEN_INT_CNTL register.

2.11 DrawingEngine2D Registers

DST_OFFSET - RW - 32 bits - [MMReg:0x1404]			
Field Name	**Bits**	**Default**	**Description**
DST_OFFSET	31:0	0x0	Destination Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of destination offset will be set to 0. NOTE: Bits 0:3 of this field are hardwired to ZERO.

Destination Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of destination offset will be set to 0.

DST_PITCH - RW - 32 bits - [MMReg:0x1408]			
Field Name	**Bits**	**Default**	**Description**
DST_PITCH	13:0	0x0	Pitch in units of 64 bytes, 0 to 16320 bytes across. DST_TILE 31: 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 2: microtiled 3: tiled and microtiled Write bits 31: 30 to E2_DST_TILE, 29: 22 to E2_DST_PITCH and 21: 0 to E2_DST_OFFSET NOTE: Bits 0:3 of this field are hardwired to ZERO.

Pitch in units of 64 bytes, 0 to 16320 bytes across. DST_TILE 31: 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 2: microtiled 3: tiled and microtiled Write bits 31: 30 to E2_DST_TILE, 29: 22 to E2_DST_PITCH and 21: 0 to E2_DST_OFFSET

DST_WIDTH - RW - 32 bits - [MMReg:0x140C]			
Field Name	**Bits**	**Default**	**Description**
DST_WIDTH	13:0	none	Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete

Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete

DST_HEIGHT - RW - 32 bits - [MMReg:0x1410]			
Field Name	**Bits**	**Default**	**Description**
DST_HEIGHT	13:0	none	Destination Height Write 15: 0 to E2_DST_Y, Write 31: 16 to E2_DST_HEIGHT

Destination Height Write 15: 0 to E2_DST_Y, Write 31: 16 to E2_DST_HEIGHT

SRC_X - RW - 32 bits - [MMReg:0x1414]			
Field Name	**Bits**	**Default**	**Description**
SRC_X	13:0	none	Source X coordinate: range -8192 to 8191

Source X coordinate: range -8192 to 8191

SRC_Y - RW - 32 bits - [MMReg:0x1418]			
Field Name	**Bits**	**Default**	**Description**
SRC_Y	13:0	none	Source Y coordinate: range -8192 to 8191 Write 13: 0 to E2_SRC_X, Write 29: 16 to E2_SRC_Y

Source Y coordinate: range -8192 to 8191 Write 13: 0 to E2_SRC_X, Write 29: 16 to E2_SRC_Y

DST_X - RW - 32 bits - [MMReg:0x141C]			
Field Name	**Bits**	**Default**	**Description**
DST_X	13:0	none	Destination X

Destination X

DST_Y - RW - 32 bits - [MMReg:0x1420]			
Field Name	**Bits**	**Default**	**Description**
DST_Y	13:0	none	Destination Y

Destination Y

SRC_PITCH_OFFSET - W - 32 bits - [MMReg:0x1428]			
Field Name	**Bits**	**Default**	**Description**
SRC_OFFSET *(mirror bits 10:31 of SRC_OFFSET:SRC_OFFSET)*	21:0	none	Source Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of source offset will be set to 0.
SRC_PITCH *(mirror bits 6:13 of SRC_PITCH:SRC_PITCH)*	29:22	none	Pitch in units of 64 bytes, 0 to 16320 bytes across. SRC_TILE 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) Write bit 30 to E2_SRC_TILE, 29: 22 to E2_SRC_PITCH and 21: 0 to E2_SRC_OFFSET
SRC_TILE *(mirror of SRC_TILE:SRC_TILE)*	30	none	0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) Write 0 to E2_SRC_TILE SRC_PITCH_OFFSET[W/R]

No description available for this register.

DST_PITCH_OFFSET - W - 32 bits - [MMReg:0x142C]			
Field Name	**Bits**	**Default**	**Description**
DST_OFFSET *(mirror bits 10:31 of DST_OFFSET:DST_OFFSET)*	21:0	none	Destination Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of destination offset will be set to 0.
DST_PITCH *(mirror bits 6:13 of DST_PITCH:DST_PITCH)*	29:22	none	Pitch in units of 64 bytes, 0 to 16320 bytes across. DST_TILE 31: 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 2: microtiled 3: tiled and microtiled Write bits 31: 30 to E2_DST_TILE, 29: 22 to E2_DST_PITCH and 21: 0 to E2_DST_OFFSET
DST_TILE *(mirror of DST_TILE:DST_TILE)*	31:30	none	0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 2: microtiled 3: tiled and microtiled Write 9: 0 to E2_DST_TILE DST_PITCH_OFFSET[W/R]

No description available for this register.

DEFAULT_PITCH_OFFSET - RW - 32 bits - [MMReg:0x16E0]			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_OFFSET	21:0	none	Default Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of default offset will be set to 0.

DEFAULT_PITCH	29:22	none	Pitch in units of 64 bytes, 0 to 16320 bytes across. DEFAULT_TILE 31: 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 2: microtiled(destination only) 3: tiled and microtiled(destination only) Write bits 31: 30 to E2_DEFAULT_TILE, 29: 22 to E2_DEFAULT_PITCH and 21: 0 to E2_DEFAULT_OFFSET
DEFAULT_TILE	31:30	none	

No description available for this register.

DEFAULT2_PITCH_OFFSET - RW - 32 bits - [MMReg:0x16F8]			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_OFFSET	21:0	none	
DEFAULT_PITCH	29:22	none	
DEFAULT_TILE	31:30	none	

No description available for this register.

SRC_Y_X - W - 32 bits - [MMReg:0x1434]			
Field Name	**Bits**	**Default**	**Description**
SRC_X *(mirror of SRC_X:SRC_X)*	13:0	none	Source X coordinate: range -8192 to 8191
SRC_Y *(mirror of SRC_Y:SRC_Y)*	29:16	none	Source Y coordinate: range -8192 to 8191 Write 13: 0 to E2_SRC_X, Write 29: 16 to E2_SRC_Y

(aliased to SRC_X, SRC_Y) [W] (MM: 5_0D)

DST_Y_X - W - 32 bits - [MMReg:0x1438]			
Field Name	**Bits**	**Default**	**Description**
DST_X *(mirror of DST_X:DST_X)*	13:0	none	Destination X
DST_Y *(mirror of DST_Y:DST_Y)*	29:16	none	Destination Y

No description available for this register.

DST_HEIGHT_WIDTH - W - 32 bits - [MMReg:0x143C]			
Field Name	**Bits**	**Default**	**Description**
DST_WIDTH *(mirror of DST_WIDTH:DST_WIDTH)*	13:0	none	Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete
DST_HEIGHT *(mirror of DST_HEIGHT:DST_HEIGHT)*	29:16	none	Destination Height Write 15: 0 to E2_DST_Y, Write 31: 16 to E2_DST_HEIGHT

[W] Destination Height and Width

DP_GUI_MASTER_CNTL - W - 32 bits - [MMReg:0x146C]			
Field Name	**Bits**	**Default**	**Description**
GMC_SRC_PITCH_OFFSET_CNTL	0	none	If 0: use E2_DEFAULT_OFFSET for source offset use E2_DEFAULT_PITCH for source pitch use E2_DEFAULT_TILE for source tiling if 1: use E2_SRC_OFFSET for source offset use E2_SRC_PITCH for source pitch use E2_SRC_TILE for source tiling 0=(SRC_OFFSET, SRC_PITCH) = (DEFAULT_PITCH_OFFSET) 1=Leave Alone
GMC_DST_PITCH_OFFSET_CNTL	1	none	If 0: use E2_DEFAULT_OFFSET for destination offset use E2_DEFAULT_PITCH for destination pitch use E2_DEFAULT_TILE for destination tiling if 1: use E2_DST_OFFSET for destination offset use E2_DST_PITCH for destination pitch use E2_DST_TILE for destination tiling GMC_SRC_CLIPPING If 0: use E2_DEFAULT_SCISSOR_BOTTOM for source scissor bottom use E2_DEFAULT_SCISSOR_RIGHT for source scissor right if 1: use E2_SRC_SCISSOR_BOTTOM for source scissor bottom use E2_SRC_SCISSOR_RIGHT for source scissor right 0=(DST_OFFSET, DST_PITCH) = (DEFAULT_PITCH_OFFSET) 1=Leave Alone
GMC_SRC_CLIPPING	2	none	0=(SC_RIGHT, SC_BOTTOM) = (DEFAULT_SC_BOTTOM_RIGHT) 1=no default

GMC_DST_CLIPPING	3	none	If 0: use E2_DEFAULT_SCISSOR_BOTTOM for destination scissor bottom use E2_DEFAULT_SCISSOR_RIGHT for destination scissor right use 0 for destination scissor top use 0 for destination scissot left if 1: use E2_DST_SCISSOR_BOTTOM for destination scissor bottom use E2_DST_SCISSOR_RIGHT for destination scissor right use E2_DST_SCISSOR_TOP for destination scissor top use E2_DST_SCISSOR_LEFT for destination scissor left GMC_BRUSH_DATATYPE Write to E2_BAGU_BRUSHTYPE 0=(SC_LEFT, SC_TOP) = (0,0), (SC_BOTTOM, SC_RIGHT) = DEF_SC_BOTTOM_RIGHT) 1=no default
GMC_BRUSH_DATATYPE *(mirror of DP_DATATYPE:DP_BRUSH_DATATYPE)*	7:4	none	0=8X8 mono pattern (expanded to frgd, bkgd) 1=8X8 mono pattern (expanded to frgd, leave_alone) 2=RESERVED 3=RESERVED 4=RESERVED 5=RESERVED 6=32X1 mono pattern for lines (expanded to frgd, bkgd) 7=32X1 mono pattern for lines (expanded to frgd,leave_alone) 8=RESERVED 9=RESERVED 10=8X8 color (pixel type same as DST) 11=RESERVED 12=RESERVED 13=solid color for blits (use frgd) 14=solid color for lines (use frgd) 15=solid color for blits (use frgd)
GMC_DST_DATATYPE *(mirror of DP_DATATYPE:DP_DST_DATATYPE)*	11:8	none	0=RESERVED 1=RESERVED 2=8 bpp pseudocolor 3=16 bpp aRGB 1555 4=16 bpp RGB 565 5=RESERVED 6=32 aRGB 8888
GMC_SRC_DATATYPE *(mirror bits 0:1 of DP_DATATYPE:DP_SRC_DATATYPE)*	13:12	none	0=mono (expanded to frgd, bkgd) 1=mono (expanded to frgd, leave_alone) 2=RESERVED 3=color (pixel type same as DST)=3
GMC_BYTE_PIX_ORDER *(mirror of DP_DATATYPE:DP_BYTE_PIX_ORDER)*	14	none	0=pixel order from MSBit to LSBit 1=pixel order from LSBit to MSBit
GMC_DEFAULT_SEL	15	none	
GMC_ROP3 *(mirror of DP_MIX:DP_ROP3)*	23:16	none	0=ROP3 function

GMC_DP_SRC_SOURCE *(mirror of DP_MIX:DP_SRC_SOURCE)*	26:24	none	2=loaded from memory (rectangular trajectory 3=loaded thru hostadata (linear trajectory) 4=loaded thru hosdata (linear trajectory & byte-aligned)
GMC_SRC_DATATYPE2 *(mirror bits 2:2 of DP_DATATYPE:DP_SRC_DATATYPE)*	27	none	0=mono (expanded to frgd, bkgd) 1=RESERVED
GMC_CLR_CMP_FCN_DIS	28	none	
GMC_WR_MSK_DIS	30	none	

[W] (MM: 5_1B)

BRUSH_Y_X - RW - 32 bits - [MMReg:0x1474]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_X	2:0	none	and BRUSH_Y to E2_BAGU_X and E2_BAGU_Y BRUSH_DATA [63: 0] [R/W] (MM: 5_20 - 5_5F)
BRUSH_Y	10:8	none	

No description available for this register.

DP_BRUSH_BKGD_CLR - RW - 32 bits - [MMReg:0x1478]			
Field Name	**Bits**	**Default**	**Description**
DP_BRUSH_BKGD_CLR	31:0	none	Background color Write to RB2D_BRUSHDATA[3] SRC_OFFSET[W/R] Source Offset

Background color Write to RB2D_BRUSHDATA[3] SRC_OFFSET[W/R] Source Offset

DP_BRUSH_FRGD_CLR - RW - 32 bits - [MMReg:0x147C]			
Field Name	**Bits**	**Default**	**Description**
DP_BRUSH_FRGD_CLR	31:0	none	Foreground color. Write to RB2D_BRUSHDATA[2] DP_BRUSH_BKGD_CLR [R/W] (MM: 5_1E)

No description available for this register.

DP_CNTL_XDIR_YDIR_YMAJOR - W - 32 bits - [MMReg:0x16D0]			
Field Name	**Bits**	**Default**	**Description**
DST_Y_DIR	15	none	0=bottom to top 1=top to bottom
DST_X_DIR	31	none	0=right to left 1=left to right

No description available for this register.

BRUSH_DATA0 - W - 32 bits - [MMReg:0x1480]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA1 - W - 32 bits - [MMReg:0x1484]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA2 - W - 32 bits - [MMReg:0x1488]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA3 - W - 32 bits - [MMReg:0x148C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA4 - W - 32 bits - [MMReg:0x1490]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA5 - W - 32 bits - [MMReg:0x1494]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA6 - W - 32 bits - [MMReg:0x1498]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA7 - W - 32 bits - [MMReg:0x149C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA8 - W - 32 bits - [MMReg:0x14A0]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA9 - W - 32 bits - [MMReg:0x14A4]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA10 - W - 32 bits - [MMReg:0x14A8]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA11 - W - 32 bits - [MMReg:0x14AC]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA12 - W - 32 bits - [MMReg:0x14B0]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA13 - W - 32 bits - [MMReg:0x14B4]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA14 - W - 32 bits - [MMReg:0x14B8]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA15 - W - 32 bits - [MMReg:0x14BC]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA16 - W - 32 bits - [MMReg:0x14C0]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA17 - W - 32 bits - [MMReg:0x14C4]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA18 - W - 32 bits - [MMReg:0x14C8]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA19 - W - 32 bits - [MMReg:0x14CC]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA20 - W - 32 bits - [MMReg:0x14D0]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA21 - W - 32 bits - [MMReg:0x14D4]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA22 - W - 32 bits - [MMReg:0x14D8]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA23 - W - 32 bits - [MMReg:0x14DC]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA24 - W - 32 bits - [MMReg:0x14E0]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA25 - W - 32 bits - [MMReg:0x14E4]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA26 - W - 32 bits - [MMReg:0x14E8]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA27 - W - 32 bits - [MMReg:0x14EC]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA28 - W - 32 bits - [MMReg:0x14F0]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA29 - W - 32 bits - [MMReg:0x14F4]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA30 - W - 32 bits - [MMReg:0x14F8]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA31 - W - 32 bits - [MMReg:0x14FC]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA32 - W - 32 bits - [MMReg:0x1500]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA33 - W - 32 bits - [MMReg:0x1504]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA34 - W - 32 bits - [MMReg:0x1508]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA35 - W - 32 bits - [MMReg:0x150C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA36 - W - 32 bits - [MMReg:0x1510]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA37 - W - 32 bits - [MMReg:0x1514]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA38 - W - 32 bits - [MMReg:0x1518]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA39 - W - 32 bits - [MMReg:0x151C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA40 - W - 32 bits - [MMReg:0x1520]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA41 - W - 32 bits - [MMReg:0x1524]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA42 - W - 32 bits - [MMReg:0x1528]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA43 - W - 32 bits - [MMReg:0x152C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA44 - W - 32 bits - [MMReg:0x1530]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA45 - W - 32 bits - [MMReg:0x1534]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA46 - W - 32 bits - [MMReg:0x1538]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA47 - W - 32 bits - [MMReg:0x153C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA48 - W - 32 bits - [MMReg:0x1540]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA49 - W - 32 bits - [MMReg:0x1544]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA50 - W - 32 bits - [MMReg:0x1548]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA51 - W - 32 bits - [MMReg:0x154C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA52 - W - 32 bits - [MMReg:0x1550]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA53 - W - 32 bits - [MMReg:0x1554]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA54 - W - 32 bits - [MMReg:0x1558]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA55 - W - 32 bits - [MMReg:0x155C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA56 - W - 32 bits - [MMReg:0x1560]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA57 - W - 32 bits - [MMReg:0x1564]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA58 - W - 32 bits - [MMReg:0x1568]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA59 - W - 32 bits - [MMReg:0x156C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA60 - W - 32 bits - [MMReg:0x1570]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA61 - W - 32 bits - [MMReg:0x1574]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA62 - W - 32 bits - [MMReg:0x1578]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

BRUSH_DATA63 - W - 32 bits - [MMReg:0x157C]			
Field Name	**Bits**	**Default**	**Description**
BRUSH_DATA	31:0	none	

No description available for this register.

DST_WIDTH_X - W - 32 bits - [MMReg:0x1588]			
Field Name	**Bits**	**Default**	**Description**
DST_X *(mirror of DST_X:DST_X)*	13:0	none	Destination X
DST_WIDTH *(mirror of DST_WIDTH:DST_WIDTH)*	29:16	none	Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete

[W] Destination X and Width coordinate

DST_HEIGHT_WIDTH_8 - W - 32 bits - [MMReg:0x158C]			
Field Name	**Bits**	**Default**	**Description**
DST_WIDTH *(mirror bits 0:7 of DST_WIDTH:DST_WIDTH)*	23:16	none	Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete
DST_HEIGHT *(mirror bits 0:7 of DST_HEIGHT:DST_HEIGHT)*	31:24	none	Destination Height Write 15: 0 to E2_DST_Y, Write 31: 16 to E2_DST_HEIGHT

[W] (Reserved) 15: 0 DST_WIDTH 23: 16 Destination width: range 0 to 256 (ZERO extent)

SRC_X_Y - W - 32 bits - [MMReg:0x1590]			
Field Name	**Bits**	**Default**	**Description**
SRC_Y *(mirror of SRC_Y:SRC_Y)*	13:0	none	Source Y coordinate: range -8192 to 8191 Write 13: 0 to E2_SRC_X, Write 29: 16 to E2_SRC_Y
SRC_X *(mirror of SRC_X:SRC_X)*	29:16	none	13: 0 Source X coordinate: range -8192 to 8191

(aliased to SRC_X, SRC_Y) [W] (MM: 5_64)

DST_X_Y - W - 32 bits - [MMReg:0x1594]			
Field Name	**Bits**	**Default**	**Description**
DST_Y *(mirror of DST_Y:DST_Y)*	13:0	none	Destination Y
DST_X *(mirror of DST_X:DST_X)*	29:16	none	Destination X

No description available for this register.

DST_WIDTH_HEIGHT - W - 32 bits - [MMReg:0x1598]			
Field Name	**Bits**	**Default**	**Description**
DST_HEIGHT *(mirror of DST_HEIGHT:DST_HEIGHT)*	13:0	none	Destination Height Write 15: 0 to E2_DST_Y, Write 31: 16 to E2_DST_HEIGHT
DST_WIDTH *(mirror of DST_WIDTH:DST_WIDTH)*	29:16	none	Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete

Destination Width and Height

DST_WIDTH_X_INCY - W - 32 bits - [MMReg:0x159C]			
Field Name	**Bits**	**Default**	**Description**
DST_X *(mirror of DST_X:DST_X)*	13:0	none	Destination X
DST_WIDTH *(mirror of DST_WIDTH:DST_WIDTH)*	29:16	none	Destination Width Note: This is an initiator register. Y is incremented at end of blit. Write 15: 0 to E2_DST_X, Write 31: 16 to E2_DST_WIDTH, then signal blit_start. E2_DST_Y = E2_DEST_Y (+/-) E2_DST_HEIGHT as function of direction after blit is complete

[W] Destination Width and X coordinate

DST_HEIGHT_Y - W - 32 bits - [MMReg:0x15A0]			
Field Name	**Bits**	**Default**	**Description**
DST_Y *(mirror of DST_Y:DST_Y)*	13:0	none	Destination Y
DST_HEIGHT *(mirror of DST_HEIGHT:DST_HEIGHT)*	29:16	none	Destination Height Write 15: 0 to E2_DST_Y, Write 31: 16 to E2_DST_HEIGHT

No description available for this register.

SRC_OFFSET - RW - 32 bits - [MMReg:0x15AC]			
Field Name	**Bits**	**Default**	**Description**
SRC_OFFSET	31:0	0x0	Source Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of source offset will be set to 0. NOTE: Bits 0:3 of this field are hardwired to ZERO.

Source Offset in Units of 1KB, 0 to (4GB-1KB). When a write to this register occurs, bits 9: 0 of source offset will be set to 0.

SRC_PITCH - RW - 32 bits - [MMReg:0x15B0]			
Field Name	**Bits**	**Default**	**Description**
SRC_PITCH	13:0	0x0	Pitch in units of 64 bytes, 0 to 16320 bytes across. SRC_TILE 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) Write bit 30 to E2_SRC_TILE, 29: 22 to E2_SRC_PITCH and 21: 0 to E2_SRC_OFFSET NOTE: Bits 0:3 of this field are hardwired to ZERO.

Pitch in units of 64 bytes, 0 to 16320 bytes across. SRC_TILE 30 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) Write bit 30 to E2_SRC_TILE, 29: 22 to E2_SRC_PITCH and 21: 0 to E2_SRC_OFFSET

CLR_CMP_CNTL - W - 32 bits - [MMReg:0x15C0]			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_FCN_SRC	2:0	none	Color comparison function (Mnemonic, action): 0 = False (CMP_FALSE, always draw) 1 = True (CMP_TRUE, never draw) 2-3 = (reserved) 4 = SRC_CLR != CLR_CMP_CLR_SRC (CMP_EQ_COLOR, draw when eq) 5 = SRC_CLR = CLR_CMP_CLR_SRC (CMP_NEQ_COLOR, draw when neq) 6 = (reserved) 7 = SRC_CLR = CLR_CMP_CLR_SRC (CMP_EQ_FLIP, flip using expanded SRC_FRGD_CLR as flip mask when eq) (Reserved) 7: 3 CLR_CMP_FCN_DST 10: 8 Color comparison function (Mnemonic, action): 0 = False (CMP_FALSE, always draw) 1 = True (CMP_TRUE, never draw) 2-3 = (reserved) 4 = DST_CLR != CLR_CMP_CLR_DST (CMP_NEQ_COLOR, draw when eq) 5 = DST_CLR = CLR_CMP_CLR_DST (CMP_EQ_COLOR, draw when neq) 6-7 = (reserved) (Reserved) 23: 11 CLR_CMP_SRC 25: 24 Defines source for color keying: 0 = Destination 1 = Source 2 = Src and Dst 3 = reserved (Reserved) 31: 26 Write to RB2D_CLRCMP_CNTL 0=False (always draw) 1=True (never draw) 4=SRC_CLR !=CLR_CMP_CLR_SRC (draw on eq) 5=SRC_CLR = CLR_CMP_CLR_SRC (draw on neq) 7=SRC_CLR = CLR_CMP_CLR_SRC (flip on eq)
CLR_CMP_FCN_DST	10:8	none	0=False (always draw) 1=True (never draw) 4=DST_CLR !=CLR_CMP_CLR_DST (draw on eq) 5=DST_CLR = CLR_CMP_CLR_DST (draw on neq)
CLR_CMP_SRC	25:24	none	0=Destination 1=Source 2=Src and Dst

[R/W] (MM: 5_70)

CLR_CMP_CLR_SRC - W - 32 bits - [MMReg:0x15C4]			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_CLR_SRC	31:0	none	Color comparison color of source Write to RB2D_CLRCMP_CLR_SRC

Color comparison color of source Write to RB2D_CLRCMP_CLR_SRC

CLR_CMP_CLR_DST - W - 32 bits - [MMReg:0x15C8]			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_CLR_DST	31:0	none	Color comparison color of destination Write to RB2D_CLRCMP_CLR_DST

Color comparison color of destination Write to RB2D_CLRCMP_CLR_DST

CLR_CMP_MSK - W - 32 bits - [MMReg:0x15CC]			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_MSK	31:0	none	Color comparison color Write to RB2D_CLRCMP_MSK DEFAULT_PITCH_OFFSET[W/R]

No description available for this register.

DP_DST_ENDIAN - W - 32 bits - [MMReg:0x15D0]			
Field Name	**Bits**	**Default**	**Description**
DST_ENDIAN	1:0	none	0=no swap 1=16 bit swap 2=32 bit swap 3=half dword swap

00 = No swap (the default after reset, for all the endian converters) 01 = 16bit swap: 0xAABBCCDD becomes 0xBBAADDCC 10 = 32bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 11 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB Write to E2_DP_DST_ENDIAN

DP_SRC_ENDIAN - W - 32 bits - [MMReg:0x15D4]			
Field Name	**Bits**	**Default**	**Description**
SRC_ENDIAN	1:0	none	0=no swap 1=16 bit swap 2=32 bit swap 3=half dword swap

00 = No swap (the default after reset, for all the endian converters) 01 = 16bit swap: 0xAABBCCDD becomes 0xBBAADDCC 10 = 32bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 11 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB Write to E2_DP_SRC_ENDIAN

DP_SRC_FRGD_CLR - RW - 32 bits - [MMReg:0x15D8]			
Field Name	**Bits**	**Default**	**Description**
DP_SRC_FRGD_CLR	31:0	none	Foreground color. When color compare src eq flip is enabled, a '1' in bit location n means enable flipping on bit n. Write to E2_SRC_FRGD_CLR, And write to RB2D_CLRCMP_FLIPE

No description available for this register.

DP_SRC_BKGD_CLR - RW - 32 bits - [MMReg:0x15DC]			
Field Name	**Bits**	**Default**	**Description**
DP_SRC_BKGD_CLR	31:0	none	Background color. Write to E2_SRC_BKGD_CLR

Background color. Write to E2_SRC_BKGD_CLR

DST_LINE_START - RW - 32 bits - [MMReg:0x1600]			
Field Name	**Bits**	**Default**	**Description**
DST_START_X	13:0	none	Start X
DST_START_Y	29:16	none	Start Y Write 13: 0 to E2_LINE_START_X; Write 29: 16 to E2_LINE_START_Y; Write 0 to E2_LINE_PATCOUNT

No description available for this register.

DST_LINE_END - RW - 32 bits - [MMReg:0x1604]			
Field Name	**Bits**	**Default**	**Description**
DST_END_X	13:0	none	End X
DST_END_Y	29:16	none	End Y Write 15: 0 to E2_LINE_END_X; Write 31: 16 to E2_LINE_END_Y; signal line_start; then copy E2_LINE_END_X into E2_LINE_START_X and E2_LINE_END_Y into E2_LINE_START_Y.

No description available for this register.

DST_LINE_PATCOUNT - RW - 32 bits - [MMReg:0x1608]			
Field Name	**Bits**	**Default**	**Description**
LINE_PATCOUNT	4:0	none	Line pattern counter Write 4: 0 to E2_LINE_PATCOUNT. When DST_LINE_START register is written into the line pattern counter is reset to 0. If a non-zero starting value is desired, this register must be written in after writing to DST_LINE_START register. BRUSH_Y_X [W/R] (MM: 5_1D)

No description available for this register.

SC_LEFT - RW - 32 bits - [MMReg:0x1640]			
Field Name	**Bits**	**Default**	**Description**
SC_LEFT	13:0	none	Range -8192 to 8191.

Range -8192 to 8191.

SC_RIGHT - RW - 32 bits - [MMReg:0x1644]			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT	13:0	none	Range -8192 to 8191.

Range -8192 to 8191.

SC_TOP - RW - 32 bits - [MMReg:0x1648]			
Field Name	**Bits**	**Default**	**Description**
SC_TOP	13:0	none	Range -8192 to 8191. (Reserved) Write 23: 16 to E2_DST_SCISSOR_TOP, Write 13: 0 to E2_DST_SCISSOR_LEFT

Range -8192 to 8191. (Reserved) Write 23: 16 to E2_DST_SCISSOR_TOP, Write 13: 0 to E2_DST_SCISSOR_LEFT

SC_BOTTOM - RW - 32 bits - [MMReg:0x164C]			
Field Name	**Bits**	**Default**	**Description**
SC_BOTTOM	13:0	none	Range -8192 to 8191. (Reserved) Write 23: 16 to E2_SRC_SCISSOR_BOTTOM, Write 13: 0 to E2_SRC_SCISSOR_RIGHT

Range -8192 to 8191. (Reserved) Write 23: 16 to E2_SRC_SCISSOR_BOTTOM, Write 13: 0 to E2_SRC_SCISSOR_RIGHT

SRC_SC_RIGHT - RW - 32 bits - [MMReg:0x1654]			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT	13:0	none	Range -8192 to 8191.

[R/W] (MM: 5_95)

SRC_SC_BOTTOM - RW - 32 bits - [MMReg:0x165C]			
Field Name	**Bits**	**Default**	**Description**
SC_BOTTOM	13:0	none	Range -8192 to 8191. (Reserved) Write 23: 16 to E2_SRC_SCISSOR_BOTTOM, Write 13: 0 to E2_SRC_SCISSOR_RIGHT

[R/W] (MM: 5_97)

DP_CNTL - RW - 32 bits - [MMReg:0x16C0]			
Field Name	**Bits**	**Default**	**Description**
DST_X_DIR *(mirror of DP_CNTL_XDIR_YDIR_YMAJOR:DST_X_DIR)*	0	none	Destination X0 direction 0 = right to left 1 = left to right Reserved (Must be unused ---. Used by parser to set only xdir/ydir of blits) Write 31 to E2_CNTL_DIRECTION bit 1, Write bit 15 to E2_CNTL_DIRECTION bit 0 0=right to left 1=left to right
DST_Y_DIR *(mirror of DP_CNTL_XDIR_YDIR_YMAJOR:DST_Y_DIR)*	1	none	1 Destination Y direction 0 = bottom to top 1 = top to bottom This bit is written during setup engine initiated operations. Note that this bit is assumed to be '1' for all triangles. This bit is set to '1' by a GUI_MASTER_CNTL write DST_TILE 4: 3 0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 3: tiled and microtiled Write 0: 1 to E2_CNTL_DIRECTION and 4: 3 to E2_DST_TILE 0=bottom to top 1=top to bottom

DST_TILE *(mirror of DST_TILE:DST_TILE)*	4:3	none	

[R/W] (MM: 5_B0)

DP_DATATYPE - RW - 32 bits - [MMReg:0x16C4]			
Field Name	**Bits**	**Default**	**Description**
DP_DST_DATATYPE	3:0	none	in DP_DATATYPE 0=RESERVED 1=RESERVED 2=8 bpp pseudocolor 3=16 bpp aRGB 1555 4=16 bpp RGB 565 5=RESERVED 6=32 aRGB 8888
DP_BRUSH_DATATYPE	11:8	none	0=8X8 mono pattern (expanded to frgd, bkgd) 1=8X8 mono pattern (expanded to frgd, leave_alone) 2=RESERVED 3=RESERVED 4=RESERVED 5=RESERVED 6=32X1 mono pattern for lines (expanded to frgd, bkgd) 7=32X1 mono pattern for lines (expanded to frgd,leave_alone) 8=RESERVED 9=RESERVED 10=8X8 color (pixel type same as DST) 11=RESERVED 12=RESERVED 13=solid color for blits (use frgd) 14=solid color for lines (use frgd) 15=solid color for blits (use frgd)
DP_SRC_DATATYPE	18:16	none	0=mono (expanded to frgd, bkgd) 1=mono (expanded to frgd, leave_alone) 2=RESERVED 3=color (pixel type same as DST)=3 4=RESERVED 5=8bpp, use clut to translate 6=32bpp, use clut to translate 7=Obuffer(64 bit in, 32 bit out
DP_BYTE_PIX_ORDER	30	none	0=pixel order from MSBit to LSBit 1=pixel order from LSBit to MSBit

[R/W] (MM: 5_B1)

DP_MIX - RW - 32 bits - [MMReg:0x16C8]			
Field Name	**Bits**	**Default**	**Description**
DP_SRC_SOURCE	10:8	none	2=loaded from memory (rectangular trajectory 3=loaded thru hostadata (linear trajectory) 4=loaded thru hosdata (linear trajectory & byte-aligned)
DP_ROP3	23:16	none	Windows 3.1 ROP3 code. Ternary Raster Operations 0=ROP3 function

[R/W] (MM: 5_B2) (Reserved) 7: 0 DP_SRC_SOURCE 10: 8 Src source: 2 = loaded from memory (rectangular trajectory) 3 = loaded thru hostdata (linear trajectory) 4 = loaded thru hostdata (linear trajectory & byte-aligned) Note that during 3D/Scaler Operations (whenever SCALE_3D_FCN is non-zero) the DP_SRC_SOURCE field is ignored and data is always loaded from the 3D/Scaler pipeline

DP_WRITE_MSK - W - 32 bits - [MMReg:0x16CC]			
Field Name	**Bits**	**Default**	**Description**
DP_WRITE_MSK	31:0	none	Write mask Write to RB2D_WRITEMASK

Write mask Write to RB2D_WRITEMASK

SC_TOP_LEFT - W - 32 bits - [MMReg:0x16EC]			
Field Name	**Bits**	**Default**	**Description**
SC_LEFT *(mirror of SC_LEFT:SC_LEFT)*	13:0	none	Range -8192 to 8191.
SC_TOP *(mirror of SC_TOP:SC_TOP)*	29:16	none	Range -8192 to 8191. (Reserved) Write 23: 16 to E2_DST_SCISSOR_TOP, Write 13: 0 to E2_DST_SCISSOR_LEFT

No description available for this register.

SC_BOTTOM_RIGHT - W - 32 bits - [MMReg:0x16F0]			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT *(mirror of SC_RIGHT:SC_RIGHT)*	13:0	none	Range -8192 to 8191.
SC_BOTTOM *(mirror of SC_BOTTOM:SC_BOTTOM)*	29:16	none	Range -8192 to 8191. (Reserved) Write 23: 16 to E2_SRC_SCISSOR_BOTTOM, Write 13: 0 to E2_SRC_SCISSOR_RIGHT

(aliased to SC_BOTTOM, SC_RIGHT) [W] (MM: 5_BC)

DEFAULT_SC_BOTTOM_RIGHT - RW - 32 bits - [MMReg:0x16E8]			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_SC_RIGHT	13:0	none	Range -8192 to 8191.
DEFAULT_SC_BOTTOM	29:16	none	Range -8192 to 8191. (Reserved) 31: 10 Write 13: 0 to E2_DEFAULT_SCISSOR_RIGHT Write 29: 16 to E2_DEFAULT_SCISSOR_BOTTOM

[R/W] (MM: 5_BA)

DEFAULT2_SC_BOTTOM_RIGHT - RW - 32 bits - [MMReg:0x16DC]			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_SC_RIGHT	13:0	none	
DEFAULT_SC_BOTTOM	29:16	none	

No description available for this register.

SRC_SC_BOTTOM_RIGHT - W - 32 bits - [MMReg:0x16F4]			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT *(mirror of SRC_SC_RIGHT:SC_RIGHT)*	13:0	none	Range -8192 to 8191.
SC_BOTTOM *(mirror of SRC_SC_BOTTOM:SC_BOTTOM)*	29:16	none	Range -8192 to 8191. (Reserved) Write 23: 16 to E2_SRC_SCISSOR_BOTTOM, Write 13: 0 to E2_SRC_SCISSOR_RIGHT

(aliased to SRC_SC_BOTTOM, SRC_SC_RIGHT) [W] (MM: 5_BD)

DST_TILE - RW - 32 bits - [MMReg:0x1700]			
Field Name	**Bits**	**Default**	**Description**
DST_TILE	1:0	none	

0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) 2: microtiled 3: tiled and microtiled Write 9: 0 to E2_DST_TILE DST_PITCH_OFFSET[W/R]

SRC_TILE - RW - 32 bits - [MMReg:0x1704]			
Field Name	**Bits**	**Default**	**Description**
SRC_TILE	0	none	0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) Write 0 to E2_SRC_TILE SRC_PITCH_OFFSET[W/R]

0: linear 1: tiled (tiled imposes additional restrictions on pitch and offset) Write 0 to E2_SRC_TILE SRC_PITCH_OFFSET[W/R]

SRC_CLUT_ADDRESS - RW - 32 bits - [MMReg:0x1780]			
Field Name	**Bits**	**Default**	**Description**
SRC_CLUT_ADDRESS	7:0	none	to E2_SRC_CLUT_ADDRESS

to E2_SRC_CLUT_ADDRESS

SRC_CLUT_DATA - W - 32 bits - [MMReg:0x1784]			
Field Name	**Bits**	**Default**	**Description**
SRC_CLUT_DATA	31:0	none	CLUT data. Read

CLUT data. Read

SRC_CLUT_DATA_RD - R - 32 bits - [MMReg:0x1788]			
Field Name	**Bits**	**Default**	**Description**
SRC_CLUT_DATA	31:0	none	CLUT data. Read

No description available for this register.

HOST_DATA0 - RW - 32 bits - [MMReg:0x17C0]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA	31:0	none	

No description available for this register.

HOST_DATA1 - W - 32 bits - [MMReg:0x17C4]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA2 - W - 32 bits - [MMReg:0x17C8]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA3 - W - 32 bits - [MMReg:0x17CC]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA4 - W - 32 bits - [MMReg:0x17D0]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA5 - W - 32 bits - [MMReg:0x17D4]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA6 - W - 32 bits - [MMReg:0x17D8]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA7 - W - 32 bits - [MMReg:0x17DC]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	

No description available for this register.

HOST_DATA_LAST - W - 32 bits - [MMReg:0x17E0]			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA_LAST *(mirror of HOST_DATA0:HOST_DATA)*	31:0	none	Host Data Last. This register is here for R128 compatibility. Write to E2_HOST_DATA

No description available for this register.

DP_XOP - W - 32 bits - [MMReg:0x17F8]			
Field Name	**Bits**	**Default**	**Description**
XOP_A	7:0	none	Depends on XOP selected by XOP_OP
XOP_B	15:8	none	Depends on XOP selected by XOP_OP

XOP_C	23:16	none	Depends on XOP selected by XOP_OP
XOP_OP	25:24	none	Extended operations code, overrides ROP 0 - Normal 1 - D3D/OpenGL blend 2 - Alpha blend (32-bit only) (Reserved) 31: 26 Write to RB2D_XOP

[R/W] (MM: 0x0244)

DSTCACHE_MODE - W - 32 bits - [MMReg:0x1710]			
Field Name	**Bits**	**Default**	**Description**
DSTCACHE_MODE	31:0	none	

No description available for this register.

DSTCACHE_CTLSTAT - W - 32 bits - [MMReg:0x1714]			
Field Name	**Bits**	**Default**	**Description**
DSTCACHE_CTLSTAT	31:0	none	

No description available for this register.

PD2_DATA - W - 32 bits - [MMReg:0x1718]			
Field Name	**Bits**	**Default**	**Description**
PD2_DATA	31:0	none	

No description available for this register.

2.12 Idct Registers

IDCT_RUNS - RW - 32 bits - [MMReg:0x1F80]			
Field Name	**Bits**	**Default**	**Description**
IDCT_RUNS_3	7:0	0x0	
IDCT_RUNS_2	15:8	0x0	
IDCT_RUNS_1	23:16	0x0	
IDCT_RUNS_0	31:24	0x0	

No description available for this register.

IDCT_LEVELS - RW - 32 bits - [MMReg:0x1F84]			
Field Name	**Bits**	**Default**	**Description**
IDCT_LEVEL_HI	15:0	0x0	
IDCT_LEVEL_LO	31:16	0x0	

No description available for this register.

IDCT_CONTROL - RW - 32 bits - [MMReg:0x1FBC]			
Field Name	**Bits**	**Default**	**Description**
IDCT_CTL_LUMA_RD_FORMAT	1:0	0x0	0=0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 1=0,2,4,6,8,10,12,14,1,3,5,7,9,11,13,15 2=0,8,1,9,2,10,3,11,4,12,5,13,6,14,7,15 3=Reserved
IDCT_CTL_CHROMA_RD_FORMAT	3:2	0x0	0=0,1,2,3,4,5,6,7 1=0,2,4,6,1,3,5,7 2=Reserved 3=Reserved
IDCT_CTL_SCAN_PATTERN	4	0x0	0=Scan pattern0 1=scan pattern 1
IDCT_CTL_INTRA	5	0x0	1=Clamp IDCT Result 0..255 after adding 128
IDCT_CTL_FLUSH	6	0x0	1=Flush last block in MB trigger
IDCT_CTL_PASSTHRU	7	0x0	1=IDCT_Pass Through Mode
IDCT_CTL_SW_RESET	8	0x0	1=SW_Reset Trigger

IDCT_CTL_CONSTREQ	9	0x0	1=Constant IDCT_REQ
IDCT_CTL_SCRAMBLE	10	0x0	1=Scrambled RUNS

No description available for this register.

IDCT_AUTH_CONTROL - RW - 32 bits - [MMReg:0x1F88]			
Field Name	**Bits**	**Default**	**Description**
CONTROL_BITS	31:0	0x0	

No description available for this register.

IDCT_AUTH - RW - 32 bits - [MMReg:0x1F8C]			
Field Name	**Bits**	**Default**	**Description**
AUTH	31:0	0x0	

No description available for this register.

CP_RB_CNTL - RW - 32 bits - [MMReg:0x704]			
Field Name	**Bits**	**Default**	**Description**
RB_BUFSZ	5:0	0x0	Ring Buffer Size. This size is expressed in log2 of the actual size (in 64-bit 'quadwords'). For example, for a buffer of 1024 quadwords, you would program this field to 10(decimal). Default = 0
RB_BLKSZ	13:8	0x0	Ring Buffer Block Size. This defines the number of quad-words that the Command Processor will read between updates to the host's copy of the Read Pointer. This size is expressed in log2 of the actual size (in 64-bit 'quadwords'). For example, for a block of 1024 quadwords, you would program this field to 10(decimal). Default = 0
BUF_SWAP	17:16	0x0	Endian Swap Control for Ring Buffer and Indirect Buffer. Only affects the chip behavior if the buffer resides in system memory. 0 = No swap 1 = 16-bit swap: 0xAABBCCDD becomes 0xBBAADDCC 2 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 3 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB Default = 0
MAX_FETCH	19:18	0x0	Maximum Fetch Size for any read request that the CP makes to memory. 0 = 1 octword. (16 bytes) 1 = 2 octwords. (32 bytes) 2 = 4 octwords. (64 bytes) 3 = reserved Default = 0
RB_NO_UPDATE	27	0x0	Ring Buffer No Write to Read Pointer 0= Write to Host's copy of Read Pointer in system memory. 1= Do not write to Host's copy of Read pointer. The purpose of this control bit is to have a fall-back position if the bus-mastered write to system memory doesn't work, in which case the driver will have to read the Graphics Con-troller's copy of the Read Pointer directly, with some perfor-mance penalty. Default = 0
RB_RPTR_WR_ENA	31	0x0	Ring Buffer Read Pointer Write Transfer Enable. When set the contents of the CP_RB_RPTR_WR register is trans-ferred to the active read pointer (CP_RB_RPTR) whenever the CP_RB_WPTR register is written. Default=0 (Not Enabled).

Ring Buffer Control

CP_RB_BASE - RW - 32 bits - [MMReg:0x700]			
Field Name	**Bits**	**Default**	**Description**
RB_BASE	31:2	0x0	Ring Buffer Base. Address of the beginning of the ring buffer.

Ring Buffer Base

CP_RB_RPTR_ADDR - RW - 32 bits - [MMReg:0x70C]			
Field Name	**Bits**	**Default**	**Description**
RB_RPTR_SWAP	1:0	0x0	Swap Control to MC when writing the read pointer.
RB_RPTR_ADDR	31:2	0x0	Ring Buffer Read Pointer Address. Address of the Host's copy of the Read Pointer. CP_RB_RPTR (RO) Ring Buffer Read Pointer

Ring Buffer Read Pointer Address

CP_RB_RPTR - R - 32 bits - [MMReg:0x710]			
Field Name	**Bits**	**Default**	**Description**
RB_RPTR	22:0	0x0	Ring Buffer Read Pointer. This is an index (in dwords) of the current element being read from the ring buffer. Read Only.

Ring Buffer Read Pointer Address

CP_RB_RPTR_WR - RW - 32 bits - [MMReg:0x71C]			
Field Name	**Bits**	**Default**	**Description**
RB_RPTR_WR	22:0	0x0	Writable Ring Buffer Read Pointer. Used to set the read pointer following an ACPI event.

Writable Ring Buffer Read Pointer Address

CP_RB_WPTR - RW - 32 bits - [MMReg:0x714]			
Field Name	**Bits**	**Default**	**Description**
RB_WPTR	22:0	0x0	Ring Buffer Write Pointer. This is an index (in dwords) of the last known element to be written to the ring buffer (by the host).

Ring Buffer Write Pointer

CP_RB_WPTR_DELAY - RW - 32 bits - [MMReg:0x718]			
Field Name	**Bits**	**Default**	**Description**
PRE_WRITE_TIMER	27:0	0x0	Pre-Write Timer. The number of clocks that a write to the CP_RB_WPTR register will be delayed until actually taking effect. Default = 0
PRE_WRITE_LIMIT	31:28	0x0	Pre-Write Limit. The number of times that the CP_RB_WPTR register can be written (while the PRE_WRITE_TIMER has not expired) before the CP_RB_WPTR register is forced to be updated with the most recently written value. Default = 0

Ring Buffer Write Pointer Delay

CP_IB_BASE - RW - 32 bits - [MMReg:0x738]			
Field Name	**Bits**	**Default**	**Description**
IB_BASE	31:2	0x0	Indirect Buffer Base. Address of the beginning of the indirect buffer.

Indirect Buffer Base

CP_IB_BUFSZ - RW - 32 bits - [MMReg:0x73C]			
Field Name	**Bits**	**Default**	**Description**
IB_BUFSZ	22:0	0x0	Indirect Buffer Size. This size is expressed in dwords. This field is an initiator to begin fetching commands from the Indirect Buffer. Default = 0

Indirect Buffer Size

CP_CSQ_CNTL - RW - 32 bits - [MMReg:0x740]			
Field Name	**Bits**	**Default**	**Description**
CSQ_CNT_PRIMARY (R)	7:0	0x0	Count of available dwords in the queue for the Primary Stream. Read Only.
CSQ_CNT_INDIRECT (R)	15:8	0x0	Count of available dwords in the queue for the Indirect Stream. Read Only.
CSQ_MODE	31:28	0x0	Command Stream Queue Mode. Controls whether each command stream is enabled, and whether it is in push mode (Programmed I/O), or pull mode (Bus-Master). Encodings are chosen to be compatible with Rage128. 0= Primary Disabled, Indirect Disabled. 1= Primary PIO, Indirect Disabled. 2= Primary BM, Indirect Disabled. 3,5,7= Primary PIO, Indirect BM. 4,6,8= Primary BM, Indirect BM. 9-14= Reserved. 15= Primary PIO, Indirect PIO Default = 0

Command Stream Queue Control

SCRATCH_UMSK - RW - 32 bits - [MMReg:0x770]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_UMSK	5:0	0x0	Update Mask for ScratchPad Registers. One bit for each of the scratchpad registers. 1 = Write the contents of the respective ScratchPad register to Memory (using the SCRATCH_ADDR register as a base) whenever that ScratchPad register is written. 0 = No write to memory. Default = 0
SCRATCH_SWAP	17:16	0x0	Update Mask for ScratchPad Registers. One bit for each of the scratchpad registers. 1 = Write the contents of the respective ScratchPad register to Memory (using the SCRATCH_ADDR register as a base) whenever that ScratchPad register is written. 0 = No write to memory. Default = 0

Micro Engine Control

SCRATCH_ADDR - RW - 32 bits - [MMReg:0x774]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_ADDR	31:5	0x0	Memory Address to which the contents of scratchpad registers should be written. Reg 0 is written to address: SCRATCH_ADDR + 0 Reg 1 is written to address: SCRATCH_ADDR + 4 Reg 2 is written to address: SCRATCH_ADDR + 8 … etc…

ScratchPad Register Address

CP_ME_CNTL - RW - 32 bits - [MMReg:0x7D0]			
Field Name	**Bits**	**Default**	**Description**
ME_STAT (R)	15:0	0x0	Status of MicroEngine internal registers. This value depends on the current value of the ME_STATMUX field. Read Only.
ME_STATMUX	20:16	0x0	Selects which status is to be returned on the ME_STAT field.
ME_BUSY (R)	29	0x0	Busy indicator for the MicroEngine. 0 = MicroEngine not busy. 1 = MicroEngine is active. Read Only.
ME_MODE	30	0x1	Run-Mode of MicroEngine. 0 = Single-Step Mode. 1 = Free-running Mode. Default = 1
ME_STEP (W)	31	0x0	Step the MicroEngine by one instruction. Writing a '1' to this field causes the MicroEngine to step by one instruction, if and only if the ME_MODE bit is a '0'. Write Only.

Micro Engine Control

CP_ME_RAM_ADDR - RW - 32 bits - [MMReg:0x7D4]			
Field Name	**Bits**	**Default**	**Description**
ME_RAM_ADDR	7:0	0x0	MicroEngine RAM Address (Write Mode) Writing this register puts the RAM access circuitry into 'Write Mode' , which allows the address to auto-increment as data is written into the RAM.

MicroEngine RAM Address

CP_ME_RAM_RADDR - W - 32 bits - [MMReg:0x7D8]			
Field Name	**Bits**	**Default**	**Description**
ME_RAM_RADDR *(mirror of CP_ME_RAM_ADDR:ME_RAM_ADDR)*	7:0	0x0	MicroEngine RAM Address (Read Mode) Writing this register puts the RAM access circuitry into 'Read Mode' , which allows the address to auto-increment as data is read from the RAM. Write Only.

MicroEngine RAM Read Address

CP_ME_RAM_DATAH - RW - 32 bits - [MMReg:0x7DC]			
Field Name	**Bits**	**Default**	**Description**
ME_RAM_DATAH	5:0	0x0	MicroEngine RAM Data High Used to load the MicroEngine RAM.

MicroEngine RAM Data High

CP_ME_RAM_DATAL - RW - 32 bits - [MMReg:0x7E0]			
Field Name	**Bits**	**Default**	**Description**
ME_RAM_DATAL	31:0	0x0	MicroEngine RAM Data Low Used to load the MicroEngine RAM.

MicroEngine RAM Data Low

CP_CSQ_ADDR - W - 32 bits - [MMReg:0x7F0]			
Field Name	**Bits**	**Default**	**Description**
CSQ_ADDR	9:2	0x0	Address into the Command Stream Queue which is to be read from. Used for debug, to read the contents of the Command Stream Queue.

(WO) Command Stream Queue Address

CP_CSQ_DATA - R - 32 bits - [MMReg:0x7F4]			
Field Name	**Bits**	**Default**	**Description**
CSQ_DATA	31:0	0x0	Data from the Command Stream Queue, from location pointed to by the CP_CSQ_ADDR register. Used for debug, to read the contents of the Command Stream Queue.

(RO) Command Stream Queue Data

CP_CSQ_STAT - R - 32 bits - [MMReg:0x7F8]			
Field Name	**Bits**	**Default**	**Description**
CSQ_RPTR_PRIMARY	7:0	0x0	Current Read Pointer into the Primary Queue. Default = 0.
CSQ_WPTR_PRIMARY	15:8	0x0	Current Write Pointer into the Primary Queue. Default = 0.
CSQ_RPTR_INDIRECT	23:16	0x0	Current Read Pointer into the Indirect Queue. Default = 0.
CSQ_WPTR_INDIRECT	31:24	0x0	Current Write Pointer into the Indirect Queue. Default = 0.

(RO) Command Stream Queue Status

SCRATCH_REG0 - RW - 32 bits - [MMReg:0x15E0]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_REG0	31:0	0x0	ScratchPad Register.

ScratchPad Register.

SCRATCH_REG1 - RW - 32 bits - [MMReg:0x15E4]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_REG1	31:0	0x0	ScratchPad Register.

ScratchPad Register.

SCRATCH_REG2 - RW - 32 bits - [MMReg:0x15E8]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_REG2	31:0	0x0	ScratchPad Register.

ScratchPad Register.

SCRATCH_REG3 - RW - 32 bits - [MMReg:0x15EC]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_REG3	31:0	0x0	ScratchPad Register.

ScratchPad Register.

SCRATCH_REG4 - RW - 32 bits - [MMReg:0x15F0]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_REG4	31:0	0x0	ScratchPad Register.

ScratchPad Register.

SCRATCH_REG5 - RW - 32 bits - [MMReg:0x15F4]			
Field Name	**Bits**	**Default**	**Description**
SCRATCH_REG5	31:0	0x0	ScratchPad Register.

ScratchPad Register.

CP_CSQ_APER_PRIMARY - RW - 32 bits - [MMReg:0x1000-0x11FC]			
Field Name	**Bits**	**Default**	**Description**
CP_CSQ_APER_PRIMARY	31:0	0x0	

No description available for this register.

CP_CSQ_APER_INDIRECT - RW - 32 bits - [MMReg:0x1300-0x13FC]			
Field Name	**Bits**	**Default**	**Description**
CP_CSQ_APER_INDIRECT	31:0	0x0	

No description available for this register.

CP_DEBUG - RW - 32 bits - [MMReg:0x7EC]			
Field Name	**Bits**	**Default**	**Description**
CP_DEBUG	31:0	0x0	Reserved for future debug considerations

Debug Register

CP_STAT - R - 32 bits - [MMReg:0x7C0]			
Field Name	**Bits**	**Default**	**Description**
MRU_BUSY	0	0x0	Memory Read Unit is Busy
MWU_BUSY	1	0x0	Memory Write Unit is Busy
RSIU_BUSY	2	0x0	Register Server Interface is Busy
RCIU_BUSY	3	0x0	Register Client Interface is Busy
CSF_PRIMARY_BUSY	9	0x0	Command Stream Fetcher is Busy operating on the Primary Stream
CSF_INDIRECT_BUSY	10	0x0	Command Stream Fetcher is Busy operating on the Indirect Stream
CSQ_PRIMARY_BUSY	11	0x0	Command Stream Queue is Busy operating on the Primary Stream
CSQ_INDIRECT_BUSY	12	0x0	Command Stream Queue is Busy operating on the Indirect Stream
CSI_BUSY	13	0x0	Command Stream Interpreter is Busy
GUIDMA_BUSY	28	0x0	GUI DMA Engine is Busy
reserved	29		reserved
CMDSTRM_BUSY	30	0x0	Command Processor's Command Stream is Busy
CP_BUSY	31	0x0	Any Block in the CP is Busy

Command Processor Status Register

2.14 dma Registers

DMA_GUI_TABLE_ADDR - W - 32 bits - [MMReg:0x780]			
Field Name	**Bits**	**Default**	**Description**
CP_SYNC	0	0x0	Synchronize with the Command Processor. Indicates that the MicroEngine cannot proceed to write anything to the register backbone while this DMA is running.
TABLE_ADDR	31:4	0x0	Memory Address of the most recently active descriptor.

(WO) GUI DMA Engine Descriptor Table Address

DMA_GUI_SRC_ADDR - R - 32 bits - [MMReg:0x784]			
Field Name	**Bits**	**Default**	**Description**
SRC_ADDR	31:0	0x0	Memory Address or Register-space Address where Source data begins, for the currently active descriptor that is being processed by the DMA engine. Read Only.

GUI DMA Engine Source Address

DMA_GUI_DST_ADDR - R - 32 bits - [MMReg:0x788]			
Field Name	**Bits**	**Default**	**Description**
DST_ADDR	31:0	0x0	Memory Address or Register-space Address where Source data is being written, for the currently active descriptor that is being processed by the DMA engine. Read Only.

GUI DMA Engine Destination Address

DMA_GUI_COMMAND - R - 32 bits - [MMReg:0x78C]			
Field Name	**Bits**	**Default**	**Description**
BYTE_COUNT	20:0	0x0	Number of Bytes remaining to be transferred from Source to Destination.

SRC_SWAP	23:22	0x0	Source Endian Swap Control. 0 = No swap 1 = 16-bit swap: 0xAABBCCDD becomes 0xBBAADDCC 2 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 3 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB
DST_SWAP	25:24	0x0	Destination Endian Swap Control. 0 = No swap 1 = 16-bit swap: 0xAABBCCDD becomes 0xBBAADDCC 2 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 3 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB
SAS	26	0x0	Source Address Space. 0 = Source Address is a memory-space address 1 = Source Address is a register-space address
DAS	27	0x0	Destination Address Space. 0 = Destination Address is a memory-space address 1 = Destination Address is a register-space address
SAIC	28	0x0	Source Address Increment Control. 0 = Increment the internal Source Address with each data transfer. 1 = No increment.
DAIC	29	0x0	Destination Address Increment Control. 0 = Increment the internal Destination Address with each data transfer. 1 = No increment.
INTDIS	30	0x0	Interrupt Disable. This value is a don't care if the EOL bit is '0'. 1= Disable the EndOfList interrupt. 0= Don't disable interrupt.
EOL	31	0x0	End Of List. Indicates that the currently active descriptor is the last one in the Descriptor Table.

(RO)

DMA_GUI_STATUS - RW - 32 bits - [MMReg:0x790]			
Field Name	**Bits**	**Default**	**Description**
DTAQ_AVAIL (R)	4:0	0x0	The number of available entries in the Descriptor Table Address Queue. Read Only.
LAST_TABLE_NUM (R)	11:8	0x0	This is a pointer into the Descriptor Table Address Queue, indicating which queue entry was the last one to be written. Read Only.

CURRENT_TABLE_NUM (R)	15:12	0x0	This is a pointer into the Descriptor Table Address Queue, indicating which queue entry the DMA engine is currently processing. Read Only.
ABORT_EN	20	0x0	1 = Abort the Descriptor Table Address Queue entry pointed to by the ABORT_TABLE_NUM field. If (ABORT_TABLE_NUM= = CURRENT_TABLE_NUM) the DMA engine freezes, to allow the host the opportunity to read the status. Otherwise, the DMA engine proceeds with normal operation on current descriptor. 0 = No Abort. If (ABORT_TABLE_NUM= = CURRENT_TABLE_NUM) the DMA engine starts processing the next entry in the Descriptor Table Address Queue. Otherwise, the DMA engine continues with the current operation.
ACTIVE (R)	21	0x0	Indicates that the DMA engine is currently working on a descriptor. It indicates whether the CURRENT_TABLE_NUM field is a valid entry in the Descriptor Table Address Queue. Read Only.
SWAP	23:22	0x0	Endian Swap Control for fetching the Descriptor Table. 0 = No swap 1 = 16-bit swap: 0xAABBCCDD becomes 0xBBAADDCC 2 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 3 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB

GUI DMA Engine Status

DMA_GUI_ACT_DSCRPTR - R - 32 bits - [MMReg:0x794]			
Field Name	**Bits**	**Default**	**Description**
TABLE_ADDR	31:4	0x0	Memory Address of the most recently active descriptor.

(RO)

GEN_INT_CNTL - RW - 32 bits - [IOReg,MMReg:0x40]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_MASK (DISPLAY)	0	0x0	Vertical blank interrupt mask. 0=Disable 1=Enable
CRTC_VLINE_MASK (DISPLAY)	1	0x0	Vertical line interrupt mask. 0=Disable 1=Enable
CRTC_VSYNC_MASK (DISPLAY)	2	0x0	Vertical sync interrupt mask. 0=Disable 1=Enable

SNAPSHOT_MASK (DISPLAY)	3	0x0	Snapshot interrupt mask. 0=Disable 1=Enable
FP_DETECT_MASK (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt mask. 0=Disable 1=Enable
CRTC2_VLINE_MASK (DISPLAY)	5	0x0	0=Disable 1=Enable
CRTC2_VSYNC_MASK (DISPLAY)	6	0x0	0=Disable 1=Enable
SNAPSHOT2_MASK (DISPLAY)	7	0x0	0=Disable 1=Enable
CRTC2_VBLANK_MASK (DISPLAY)	9	0x0	0=Disable 1=Enable
FP2_DETECT_MASK (DISPLAY)	10	0x0	0=Disable 1=Enable
VSYNC_DIFF_OVER_LIMIT_MASK (TVOUT)	11	0x0	0=Disable 1=Enable
GUI_IDLE_MASK (RBBM)	19	0x0	GUI idle interrupt mask. 0=Disable 1=Enable
SW_INT_EN (HDP)	25	0x0	Software interrupt mask. 0=Disable 1=Enable
GEYSERVILLE_MASK (VIP)	27	0x0	0=Disable 1=Enable
HDCP_AUTHORIZED_INT_MASK (DISPLAY)	28	0x0	0=Disable 1=Enable
DVI_I2C_INT_MASK (DISPLAY)	29	0x0	0=Disable 1=Enable
GUIDMA_MASK (CP)	30	0x0	GUI DMA channel interrupt mask. 0=Disable 1=Enable
reserved	31		reserved

General Interrupt Control register.
The MASK/EN fields control whether the respective status bits are enabled to drive the system interrupt pin. Even if enabled here, the interrupt line is not driven unless enabled in the PCI configuration space.

GEN_INT_STATUS - RW - 32 bits - [IOReg,MMReg:0x44]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_STAT (R) (DISPLAY)	0	0x0	Vertical blank interrupt. Set when display in vertical retrace. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VBLANK_STAT_AK (W) (DISPLAY)	0	0x0	Vertical blank interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VLINE_STAT (R) (DISPLAY)	1	0x0	Vertical line interrupt. Set on display line on programmed by the CRTC_VLINE_CRNT_VLINE.CRTC_VLINE register. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VLINE_STAT_AK (W) (DISPLAY)	1	0x0	Vertical line interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VSYNC_STAT (R) (DISPLAY)	2	0x0	Vertical sync interrupt. Set on start of VSYNC at the DAC. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_STAT_AK (W) (DISPLAY)	2	0x0	Vertical sync interrupt acknowledge. 0=No effect 1=Clear status
SNAPSHOT_STAT (R) (DISPLAY)	3	0x0	Snapshot interrupt. Set as controlled by SNAPSHOT_VIF_COUNT register. 0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT_STAT_AK (W) (DISPLAY)	3	0x0	Snapshot interrupt acknowledge. 0=No effect 1=Clear status
FP_DETECT_STAT (R) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt. Set on HPD connect or disconnect as controlled by FP_GEN_CNTL.FP_DETECT_INT_POL. 0=No event 1=Event has occurred, interrupting if enabled
FP_DETECT_STAT_AK (W) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt acknowledge. 0=No effect 1=Clear status
CRTC2_VLINE_STAT (R) (DISPLAY)	5	0x0	0=No event 1=Event has occurred, interrupting if enabled

CRTC2_VLINE_STAT_AK (W) (DISPLAY)	5	0x0	0=No effect 1=Clear status
CRTC2_VSYNC_STAT (R) (DISPLAY)	6	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VSYNC_STAT_AK (W) (DISPLAY)	6	0x0	0=No effect 1=Clear status
SNAPSHOT2_STAT (R) (DISPLAY)	7	0x0	0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT2_STAT_AK (W) (DISPLAY)	7	0x0	0=No effect 1=Clear status
CAP0_INT_ACTIVE (R) (VIP)	8	0x0	Capture port 0 has active interrupt(s). 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CRTC2_VBLANK_STAT (R) (DISPLAY)	9	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VBLANK_STAT_AK (W) (DISPLAY)	9	0x0	0=No effect 1=Clear status
FP2_DETECT_STAT (R) (DISPLAY)	10	0x0	0=No event 1=Event has occurred, interrupting if enabled
FP2_DETECT_STAT_AK (W) (DISPLAY)	10	0x0	0=No effect 1=Clear status
VSYNC_DIFF_OVER_LIMIT_STAT (R) (TVOUT)	11	0x0	0=No event 1=Event has occurred, interrupting if enabled
VSYNC_DIFF_OVER_LIMIT_STAT_AK (W) (TVOUT)	11	0x0	0=No effect 1=Clear status
GUI_IDLE_STAT (R) (RBBM)	19	0x1	GUI idle interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_STAT_AK (W) (RBBM)	19	0x0	GUI idle interrupt acknowledge. 0=No effect 1=Clear status

SW_INT (R) (HDP)	25	0x0	Software interrupt. General purpose interrupt that can only be set by software event by writing to SW_INT_SET. 0=No event 1=Event has occurred, interrupting if enabled
SW_INT_AK (W) (HDP)	25	0x0	Software interrupt acknowledge. 0=No effect 1=Clear SW_INT (set low)
SW_INT_SET (W) (HDP)	26	0x0	Software interrupt trigger. 0=No effect 1=Set SW_INT active (high)
GEYSERVILLE_STAT (R) (VIP)	27	0x0	0=No event 1=Event has occurred, interrupting if enabled
GEYSERVILLE_STAT_AK (W) (VIP)	27	0x0	0=No effect 1=Clear status
HDCP_AUTHORIZED_INT_STAT (R) (DISPLAY)	28	0x0	0=No event 1=Event has occurred, interrupting if enabled
HDCP_AUTHORIZED_INT_AK (W) (DISPLAY)	28	0x0	0=No effect 1=Clear status
DVI_I2C_INT_STAT (R) (DISPLAY)	29	0x0	0=No event 1=Event has occurred, interrupting if enabled
DVI_I2C_INT_AK (W) (DISPLAY)	29	0x0	0=No effect 1=Clear status
GUIDMA_STAT (R) (CP)	30	0x0	GUI DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUIDMA_AK (W) (CP)	30	0x0	GUI DMA channel interrupt acknowledge. 0=No effect 1=Clear status
reserved	31		reserved

General Interrupt Status register.
These fields can be polled and acknowledged even if interrupts are disabled, or the respective fields are masked in the GEN_INT_CNTL register.

2.15 control Registers

General control/status registers for the display engine

DISPLAY_BASE_ADDR - RW - 32 bits - [MMReg:0x23C]			
Field Name	**Bits**	**Default**	**Description**
DISPLAY_BASE_ADDR	31:0	0x0	Base address added to all graphics, cursor and icon requests to the internal memory controller. This should normally be set equal to MC_FB_LOCATION.MC_FB_START. It could be set to other places to allow display out of system (e.g. AGP) memory, but this is not recommended. This is a 4 Megabyte aligned base address. NOTE: Bits 0:21 of this field are hardwired to ZERO.

Primary display graphics, cursor and icon base address

DISP_OUTPUT_CNTL - RW - 32 bits - [MMReg:0xD64]			
Field Name	**Bits**	**Default**	**Description**
DISP_DAC_SOURCE	1:0	0x0	DAC data source select. When doing ratiometric expansion, this should always be set to 10. When pixel clock is above 165 MHz, this MUST be set to 00. The 11 setting is used for YPbPr direct drive of televisions. The 01 setting allows advanced color adjustment for the display, but this will rarely be used. 0=DAC output data comes directly from display merger. 1=Reserved - DAC output data comes from output linear transform unit. 2=DAC output data comes from ratiometric expansion unit. 3=Reserved - DAC output data comes from display transform unit, and DAC is in YPbPr mode.
DISP_RMX_SOURCE	8	0x0	Ratiometric expansion source select. This is normally set to 0. The 1 setting can be used to improve video brightness on panels and CRTs, but this is normally done better using the overlay contrast and gamma controls. 0=Ratiometric expansion data comes directly from display merger. 1=Ratiometric expansion data comes from output linear transform unit.

DISP_RMX_DITH_EN	10	0x1	Selects how 30 bit data is converted to 24 bits at input to ratiometric expansion unit. 0=Convert 30 bit color data to 24 bits by truncation of lower bits. 1=Convert 30 bit color data to 24 bits by 1-D dithering.
DISP_TV_EVEN_FLAG_CNTL	28:27	0x0	0=TV out even flag inverse of video field polarity. i.e. 1 for even. 1=TV out even flag equal to video field polarity. i.e. 1 for odd. 2=TV out even flag forced to 0. 3=TV out even flag forced to 1.

Display outout control register

DISP_MERGE_CNTL - RW - 32 bits - [MMReg:0xD60]			
Field Name	**Bits**	**Default**	**Description**
DISP_ALPHA_MODE	1:0	0x0	Graphics and overlay blend mode select. There are three choices of blend mode as described below. Per pixel alpha is really only for graphics modes that have alpha values embedded (32bpp aRGB8888, 16bpp aRGB1555, 16bpp aRGB4444, 16bpp alphaIndex88. When the global alphas are set to 1 (i.e. 0xFF), the key mode works like key mixing in past chips. 0=Key mode: For GRPH_VID_KEY = 0, then Pixel = DISP_GRPH_ALPHA * Primary Graphics. For KEY = 1, then Pixel = DISP_OV0_ALPHA * Overlay pixel 1=Per pixel alpha mode: See DISP_ALPHA_PREMULT and DISP_ALPHA_INV for result 2=Global alpha mode: Outside overlay window Pixel = DISP_GRPH_ALPHA * Primary Graphics. Inside overlay window Pixel = (1-DISP_OV0_ALPHA * Primary Graphics + DISP_OV0_ALPHA * Overlay pixel
DISP_ALPHA_INV	2	0x0	For use with per pixel alpha blend mode. Applys optional invertion to the alpha value extracted from the graphics surface data. 0=When DISP_ALPHA_MODE = 01, then use per pixel alpha as primary graphics opacity (PIX_ALPHA <= alpha from pixel 1=When DISP_ALPHA_MODE = 01, then use per pixel alpha as primary graphics transparency (PIX_ALPHA <= 1 - (alpha from pixel))

DISP_ALPHA_PREMULT	3	0x0	For use with per pixel alpha blend mode. Selects whether pre-multiplied alpha or non-pre-multiplied alpha mix equation is used. 0=When DISP_ALPHA_MODE = 01, then Pixel = PIX_ALPHA * graphics pixel + (1-PIX_ALPHA) * overlay pixel 1=When DISP_ALPHA_MODE = 01, then Pixel = graphics pixel + (1-PIX_ALPHA) * overlay pixel
DISP_RGB_OFFSET_EN	8	0x0	Enable for no clamping for YUV input to YUV tvout output. Should always be enabled, unless it doesn't work. When enabled, DISP_LIN_TRANS must not be in bypass mode. 0=RGB mixing pipe operates in 0 to 1 (i.e. 0..1023) limited RGB space. Overlay pixel values outside this range are clamped to be in range. 1=RGB mixing pipe operates in -1.5 to +1.5 range. No clamping occurs for YCbCr or YPbPr mode outputs. Clamping is applied for RGB mode outputs.
DISP_GRPH_ALPHA	23:16	0xff	Global graphics alpha for use in key mode and global alpha modes of DISP_ALPHA_MODE.
DISP_OV0_ALPHA	31:24	0xff	Global video alpha for use in key mode and global alpha modes of DISP_ALPHA_MODE.

Controls for mixing of graphics and video layers

DISP2_MERGE_CNTL - RW - 32 bits - [MMReg:0xD68]			
Field Name	**Bits**	**Default**	**Description**
DISP2_RGB_OFFSET_EN	8	0x0	Enable for no clamping for YUV input to YUV tvout output. Should always be enabled, unless it doesn't work. When enabled, DISP_LIN_TRANS must not be in bypass mode. 0=Compositing unit operates in normal RGB space 1=Compositing unit operates in +1536 offset RGB space

No description available for this register.

DISP_MISC_CNTL - RW - 32 bits - [MMReg:0xD00]			
Field Name	**Bits**	**Default**	**Description**
SOFT_RESET_GRPH_PP	0	0x0	0=Not reset 1=Reset graphics and cursor pixel pipe in pixel clock domain
SOFT_RESET_SUBPIC_PP	1	0x0	0=Not reset 1=Reset sub-picture pixel pipe in pixel clock domain
SOFT_RESET_OV0_PP	2	0x0	0=Not reset 1=Reset ov0scale pixel pipe in pixel clock domain

SOFT_RESET_GRPH_SCLK	4	0x0	0=Not reset 1=Reset graphics and cursor logic in system clock domain
SOFT_RESET_SUBPIC_SCLK	5	0x0	0=Not reset 1=Reset sub-picture logic in system clock domain
SOFT_RESET_OV0_SCLK	6	0x0	0=Not reset 1=Reset ov0scale logic in system clock domain
SYNC_STRENGTH	9:8	0x2	0=Minimum drive. ~74 ohms. 1=Matched impedence drive. ~63 ohms. 2=Optimal drive. ~47 ohms. 3=Maximum drive. ~42 ohms.
SYNC_PAD_FLOP_EN	10	0x0	0=do not flop SYNC signals on the pad 1=flop SYNC signals on the pad
SOFT_RESET_GRPH2_PP	12	0x0	0=Not reset 1=Reset graphics and cursor pixel pipe in pixel clock domain
SOFT_RESET_GRPH2_SCLK	15	0x0	0=Not reset 1=Reset graphics and cursor logic in system clock domain
SOFT_RESET_LVDS	16	0x0	0=Not reset 1=Reset
SOFT_RESET_TMDS	17	0x0	0=Not reset 1=Reset
SOFT_RESET_DIG_TMDS	18	0x0	0=Not reset 1=Reset
SOFT_RESET_TV	19	0x0	0=Not reset 1=Reset
PALETTE2_MEM_RD_MARGIN	23:20	0x3	
PALETTE_MEM_RD_MARGIN	27:24	0x3	
RMX_BUF_MEM_RD_MARGIN	31:28	0x3	

No description available for this register.

HTOTAL_CNTL - RW - 32 bits - CLKIND:0x9			
Field Name	**Bits**	**Default**	**Description**
HTOT_PIX_SLIP (DISPLAY)	3:0	0x0	Pixel accurate control of horizontal total. Selects the extra number of pixels to add to each display line. Valid range is 0 to 15. For VGA modes with SEQ_PCLKBY2 = 1 each increment adds one physical pixel or 1/2 a logical pixel to the line total. For 9-dot 40 column VGA text modes, it is not possible to add 16/18ths or 17/18ths of a character extra to the display line times.
HTOT_VCLK_SLIP (DISPLAY)	11:8	0x0	Reserved for future use. No affect in this ASIC.
HTOT_PPLL_SLIP (CG)	18:16	0x0	Select the number of 1/5 PPllClk phase slips to do in the PLL at every HSYNC. This is used as a sub-pixel accurate adjustment of the frame rate for TV out or video gen-lock-ing.
HTOT_CNTL_EDGE (DISPLAY) (CG)	24	0x0	Select which HSYNC edge the slip correction based on HTOT_PPLL_SLIP is done.
HTOT_CNTL_VGA_EN (DISPLAY) (CG)	28	0x0	Select if the slip controls based on HTOT_PIX_SLIP, HTOT_VCLK_SLIP and HTOT_PPLL_SLIP are enabled for VGA mode. These are always enabled for non-VGA modes when the respective fields are non-zero.

Horizontal total control. Used to fine-tune the horizontal total. This lengthens the time of each display line by sub-character and/or sub-pixel amounts. The purpose is fine adjustment of the overall frame refresh rate for applications that require it (e.g. TV output, GEN-lock to video input).

HTOTAL2_CNTL - RW - 32 bits - CLKIND:0x2E			
Field Name	**Bits**	**Default**	**Description**
HTOT2_PIX_SLIP (DISPLAY)	3:0	0x0	
HTOT2_PIX2CLK_SLIP (DISPLAY)	11:8	0x0	
HTOT2_P2PLL_SLIP (CG)	18:16	0x0	
HTOT2_CNTL_EDGE (DISPLAY) (CG)	24	0x0	

HTOT2_CNTL_UPDATE (CG)	28	0x0	

No description available for this register.

DISP_PWR_MAN - RW - 32 bits - [MMReg:0xD08]			
Field Name	**Bits**	**Default**	**Description**
DISP_PWR_MAN_D3_CRTC_EN	0	0x1	0=Resume display at previous CRTC settings on D3hot to D0 power state transistion. 1=Leave CRTC idle on D3hot to D0 power transition until BIOS/driver re-initializes.
DISP2_PWR_MAN_D3_CRTC2_EN	4	0x1	0=Resume secondary display at previous CRTC settings on D3hot to D0 power state transistion. 1=Leave secondary CRTC idle on D3hot to D0 power transition until BIOS/driver re-initializes.
DISP_PWR_MAN_DPMS	9:8	0x0	0=DPMS On. HSYNC and VSYNC not blocked. 1=DPMS Standby. HSYNC blocked, VSYNC not blocked. 2=DPMS Suspend. VSYNC blocked, HSYNC not blocked. 3=DPMS Off. HSYNC and VSYNC blocked.
DISP_D3_RST	16	0x1	0=D3hot to D0 does not reset display. 1=D3hot to D0 resets all display blocks.
DISP_D3_REG_RST	17	0x1	0=D3hot to D0 does not reset display registers. 1=D3hot to D0 resets all display registers.
DISP_D3_GRPH_RST	18	0x1	0=D3hot to D0 does not reset graphics logic. 1=D3hot to D0 resets graphics logic, but not registers.
DISP_D3_SUBPIC_RST	19	0x1	0=D3hot to D0 does not reset sub-picture logic. 1=D3hot to D0 resets sub-picture logic, but not registers.
DISP_D3_OV0_RST	20	0x1	0=D3hot to D0 does not reset ov0scale logic. 1=D3hot to D0 resets ov0scale logic, but not registers.
DISP_D1D2_GRPH_RST	21	0x1	0=D1 or D2 to D0 does not reset graphics logic. 1=D1 or D2 to D0 resets graphics logic, but not registers.
DISP_D1D2_SUBPIC_RST	22	0x1	0=D1 or D2 to D0 does not reset sub-picture logic. 1=D1 or D2 to D0 resets sub-picture logic, but not registers.
DISP_D1D2_OV0_RST	23	0x1	0=D1 or D2 to D0 does not reset ov0scale logic. 1=D1 or D2 to D0 resets ov0scale logic, but not registers.
DIG_TMDS_ENABLE_RST	24	0x1	0=Disable RST when DIG_TMDS is enabled 1=Enable RST when FP2_ON is enabled
TV_ENABLE_RST	25	0x1	0=Disable RST when TV is enabled 1=Enable RST when TV_ON is enabled

AUTO_PWRUP_EN	26	0x0	0=Disable 1=Enable

No description available for this register.

DISP_TEST_DEBUG_CNTL - RW - 32 bits - [MMReg:0xD10]			
Field Name	**Bits**	**Default**	**Description**
DISP_TEST_DISPENG	0	0x0	0=Disable 1=Enable
DISP_TEST_PALETTE	1	0x0	0=Disable 1=Enable
DISP_TEST_DAC	2	0x0	0=Disable 1=Enable
DISP_TEST_SUBPIC	3	0x0	0=Disable 1=Enable
DISP_TEST_OV0SCALE	4	0x0	0=Disable 1=Enable
DISP_TEST_CLK	5	0x0	0=Disable 1=Enable
DISP_TEST_DISP2ENG	8	0x0	0=Disable 1=Enable
DISP_TEST_PALETTE2	9	0x0	0=Disable 1=Enable
DISP_GRPH2_UNDERFLOW (R)	16	0x0	0=No secondary graphics underflow 1=Secondary Graphics buffer has underflowed since last cleared
DISP_GRPH2_UNDERFLOW_CLR (W)	16	0x0	0=Write 0, no effect 1=Write 1, clear secondary graphics buffer underflow flag
DISP_CUR2_UNDERFLOW (R)	17	0x0	0=No secondary cursor underflow 1=Secondary Cursor buffer has underflowed since last cleared
DISP_CUR2_UNDERFLOW_CLR (W)	17	0x0	0=Write 0, no effect 1=Write 1, clear secondary cursor buffer underflow flag
DISP_GRPH_UNDERFLOW (R)	24	0x0	0=No graphics underflow 1=Graphics buffer has underflowed since last cleared
DISP_GRPH_UNDERFLOW_CLR (W)	24	0x0	0=Write 0, no effect 1=Write 1, clear graphics buffer underflow flag
DISP_CUR_UNDERFLOW (R)	25	0x0	0=No cursor underflow 1=Cursor buffer has underflowed since last cleared

DISP_CUR_UNDERFLOW_CLR (W)	25	0x0	0=Write 0, no effect 1=Write 1, clear cursor buffer underflow flag
DISP_OV0_UNDERFLOW (R)	26	0x0	0=No overlay underflow 1=Overlay buffer has underflowed since last cleared
DISP_OV0_UNDERFLOW_CLR (W)	26	0x0	0=Write 0, no effect 1=Write 1, clear overlay buffer underflow flag
DISP_SUBPIC_UNDERFLOW (R)	27	0x0	0=No sub-picture underflow 1=Sub-picture buffer has underflowed since last cleared
DISP_SUBPIC_UNDERFLOW_CLR (W)	27	0x0	0=Write 0, no effect 1=Write 1, clear sub-picture buffer underflow flag
DISP_SUBPIC_FORCE_HI_PRI	28	0x0	0=Use hardware circuit to generate high priority bit for memory request 1=Force subpic to be high priority memory requester

Test and debug controls for display block.

DISP_HW_DEBUG - RW - 32 bits - [MMReg:0xD14]			
Field Name	**Bits**	**Default**	**Description**
DISP_HW_0_DEBUG	0	0x0	
DISP_HW_1_DEBUG	1	0x0	
DISP_HW_2_DEBUG	2	0x0	
DISP_HW_3_DEBUG	3	0x0	
DISP_HW_4_DEBUG	4	0x0	
CRT2_DISP1_SEL	5	0x0	Select which display controller on CRT2. Please note the field name DISP_HW_5_DEBUG is used in the design and is different from the register spec. 0=Select secondary display in CRT2 1=Select primary display in CRT2
DISP_HW_6_DEBUG	6	0x0	
DISP_HW_7_DEBUG	7	0x0	
DISP_HW_8_DEBUG	8	0x0	
DISP_HW_9_DEBUG	9	0x0	
DISP_HW_A_DEBUG	10	0x0	
DISP_HW_B_DEBUG	11	0x0	
DISP_HW_C_DEBUG	12	0x0	
DISP_HW_D_DEBUG	13	0x0	

DISP_HW_E_DEBUG	14	0x0	
DISP_HW_F_DEBUG	15	0x0	Adaptive Deinterlace Implementation. 0=Add margin for the upper current 1=Add margin for both the upper and lower current
DISP_HW_10_DEBUG	16	0x0	Overlay line buffer Overflow ECO. This ECO causes the overlay to force its line buffer read pointer to 0 at SOF. 0=Disable line buffer overflow ECO 1=Enable line buffer flow ECO
DISP_HW_11_DEBUG	17	0x1	DISP_HW_10_DEBUG must be set for this bit to take effect. This bit enables/disables read behind the scan. 0=Disable read behind the scan 1=Enable read behind the scan
DISP_HW_12_DEBUG	18	0x0	
DISP_HW_13_DEBUG	19	0x0	
DISP_HW_14_DEBUG	20	0x0	
DISP_HW_15_DEBUG	21	0x0	
DISP_HW_16_DEBUG	22	0x0	
DISP_HW_17_DEBUG	23	0x0	
DISP_HW_18_DEBUG	24	0x0	
DISP_HW_19_DEBUG	25	0x0	
DISP_HW_1A_DEBUG	26	0x0	
DISP_HW_1B_DEBUG	27	0x0	
DISP_HW_1C_DEBUG	28	0x0	
DISP_HW_1D_DEBUG	29	0x0	
DISP_HW_1E_DEBUG	30	0x0	
DISP_HW_1F_DEBUG	31	0x0	

Hardware debug bits for display block. Programmed by BIOS/driver init as needed and not touched again.

2.16 DAC Registers

DAC_CNTL - RW - 32 bits - [IOReg,MMReg:0x58]			
Field Name	**Bits**	**Default**	**Description**
DAC_RANGE_CNTL (DISPLAY)	1:0	0x2	DAC control bits. Default = 2. 0=Reserved 1=Reserved 2=PS2 Output Level 3=YPbPr output level
DAC_BLANKING (DISPLAY)	2	0x0	Controls use of DAC blanking pedestal during horizontal and vertical blanks. VGA PS2 compatible monitors expect a 0 IRE blanking pedestal.
DAC_CMP_EN (DISPLAY)	3	0x1	Control DAC comparators for analog termination checking. When enabled, the results of the three comparators are read back in the DAC_CMP_OUT_R/G/B fields. Use for the DAC_FORCE fields of DAC_EXT_CNTL is recommended for testing analog monitor connection. DAC_FORCE allows the correct 10 bit data values to be forced on the DAC channels without corrupting the TMDS or TV out images. When using the DAC comparators, be sure the comparator settling times are met by waiting at least 1us between changing the DAC_FORCE parameters and reading the comparator status bits. The comparator refence voltage is 0.440V. The recommended 10 bit DAC value for testing 75 or 37.5 ohm termination for PS2 current levels is 0x1AC. For each channel, if the comparator output is 0, it indictates the termination is 75 ohms, and therefore no monitor is attached. When the comparator output is 1, the termination is 37.5 ohms and a monitor is attached.
DAC_CMP_OUT_R (R) (DISPLAY)	4	0x0	Red channel comparator output.
DAC_CMP_OUT_G (R) (DISPLAY)	5	0x0	Green channel comparator output.
DAC_CMP_OUT_B (R) (DISPLAY)	6	0x0	Blue channel comparator output.
DAC_CMP_OUTPUT (R) (DISPLAY)	7	0x0	Logical AND of R, G & B comparator outputs.

DAC_8BIT_EN (DISPLAY)	8	0x0	Enables 8 bit DAC operation. 8 bit is normal, 6 bit used for VGA emulation. When in 6 bit writes and reads to DAC_DATA and PALETTE_DATA are affected. Writes shift 6 bits left by 2 to make 8 bits in the palette memory. Reads shift 8 bit palette data right by 2 to give 6 MSBs to the host. 0=DAC_DATA and PALETTE_DATA read/writes emulate 6 bit palette 1=DAC_DATA and PALETTE_DATA read/writes emulate 8 bit palette
DAC_4BPP_PIX_ORDER (DISPLAY)	9	0x0	Selects the order of pixel nibbles within bytes for 4 bpp extended (non-VGA) display modes.
DAC_VGA_ADR_EN (BIF)	13	0x0	Enables access of the palette (DAC) at the VGA I/O DAC addresses when in extended display modes (non-VGA, or CRTC_EXT_DISP_EN=1).
DAC_EXPAND_MODE (DISPLAY)	14	0x0	Method to expand 4,5 or 6 bit Red/Green/Blue color codes to 8 bit addresses. 0 = Zero Extension 1 = Dynamic Expansion 0=Convert 4, 5 & 6 bit colors to 8 bits by zero extension 1=Convert 4, 5 & 6 bit colors to 8 bits by dynamic expansion
DAC_PDWN (DISPLAY)	15	0x0	Power down internal DAC macro bandgap unit. Should also set the R, G & B powerdowns in the DAC_MACRO_CNTL register at the same time. Powering down the DAC does not affect the digital outputs (TV or flat panel). The DAC is automatically powered down when the PMI_POWER_STATE register is not in the D0 state. Setting all DAC_PDWN fields should save about 56 mA when PS2 output levels.
CRT_SENSE (R) (DISPLAY)	16	0x0	0=CRT Monitor Detection result - not connected 1=CRT Monitor Detection result - connected
CRT_DETECTION_ON (DISPLAY)	17	0x0	0=CRT Monitor Detection disabled 1=CRT Monitor Detection enabled
DAC_CRC_CONT_EN (DISPLAY)	18	0x0	When CRC is in one shot mode, one-and-only-one frame or field is CRCed after DAC_CRC_EN is set high. In continuous mode every frame/field is CRCed and the results are valid for one field/frame until the next set of results are ready. If a field/frames results are not read within one frame in continuous mode they are over written. In one-shot mode the results remain and can be read until DAC_CRC_EN is cleared. 0=DAC CRC runs in one shot mode. 1=DAC CRC runs in continuous mode.

DAC_CRC_EN (DISPLAY)	19	0x0	Enables the CRC signature check on the data going to the DAC macro. This is what appears on the screen, and includes graphics, HW cursor, video overlay, sub-picture, and overscan. CRC will start in next vertical blank on the first pixel of the line where VSYNC goes active, and run for one field/frame. For interlaced modes the CRC block will not start working until the beginning of a field with odd/even polarity matching the DAC_CRC_FIELD setting. 0=Reset DAC CRC. 1=Enable DAC CRC for next frame/field in one shot, or all frames/fields in continuous mode.
DAC_CRC_FIELD (DISPLAY)	20	0x0	Used only for interlaced mode CRCs. Controls which field polarity starts the CRC block after DAC_CRC_EN is set high. 0=If interlace display and one shot mode, then CRC even field only 1=If interlace display and one shot mode, then CRC odd field only
DAC_LUT_COUNTER_LIMIT (DISPLAY)	22:21	0x0	Anti-sparkle timeout. The palette circuit will search for a place in the display image to hide palette reads or writes. If no hiding location is found within this many pixels, a dot-strectch is forced to hide the cycle as best as possible. Setting this register too high in low resolution modes may cause long delays on the PCI/AGP bus. The recommended settings are best. 0=anti-sparkle timeout 3 clocks 1=anti-sparkle timeout 7 clocks 2=anti-sparkle timeout 15 clocks 3=anti-sparkle timeout 31 clocks
DAC_LUT_READ_SEL (DISPLAY)	23	0x0	Used for diagnostics only. Selects palette for HOST reads. 0=Palette reads come from main palette 1=Palette reads come from secondary palette
DAC_MASK (DISPLAY)	31:24	0xff	Mirror VGA DAC_MASK. No affect in non-VGA modes. Masks off usage of individual palette index bits before pixel index is looked-up in the palette. 0 = do not use this bit of the index 1 = use this bit of the index This is a mirror of the VGA DAC_MASK register. It only has an effect in VGA emulation modes (CRTC_EXT_DISP_EN=0), not for VESA modes or extended display modes.

General control for the RGB DAC and palette.

DAC_DATA - RW - 8 bits - VGA_IO:0x3C9			
Field Name	**Bits**	**Default**	**Description**
DAC_DATA	7:0	0x0	VGA Palette (DAC) Data. Use DAC_R_INDEX and DAC_W_INDEX to set read or write mode, and entry to access. Access order is Red, Green, Blue, and then auto-increment occurs to next entry. DAC_8BIT_EN controls whether 6 or 8 bit access.

VGA Palette (DAC) Data

DAC_MASK - RW - 8 bits - VGA_IO:0x3C6			
Field Name	**Bits**	**Default**	**Description**
DAC_MASK *(mirror of DAC_CNTL:DAC_MASK)*	7:0	0x0	Masks off usage of individual palette index bits before pixel index is looked-up in the palette. 0 = do not use this bit of the index 1 = use this bit of the index Only has an effect in VGA emulation modes (CRTC_EXT_DISP_EN=0), not for VESA modes or extended display modes.

Palette index mask for VGA emulation modes.

DAC_R_INDEX - RW - 8 bits - VGA_IO:0x3C7			
Field Name	**Bits**	**Default**	**Description**
DAC_R_INDEX	7:0	0x0	Write: Sets the index for a palette (DAC) read operation. Index auto-increments after every third read of DAC_DATA. Read: Indicates if palette in read or write mode. 0 = Palette in write mode (DAC_W_INDEX last written). 3 = Palette in read mode (DAC_R_INDEX last written). Aslo see DAC_W_INDEX.

Palette (DAC) Read Index

DAC_W_INDEX - RW - 8 bits - VGA_IO:0x3C8			
Field Name	**Bits**	**Default**	**Description**
DAC_W_INDEX	7:0	0x0	Sets the index for a palette (DAC) write operation. Index auto-increments after every third write of DAC_DATA. Aslo see DAC_R_INDEX.

Palette (DAC) Write Index

DAC_EXT_CNTL - RW - 32 bits - [MMReg:0x280]			
Field Name	**Bits**	**Default**	**Description**
DAC2_FORCE_BLANK_OFF_EN	0	0x0	0=Normal DAC2 BLANK functionality. 1=DAC2 BLANK forced off. Use this setting with DAC FORCE logic to detect CRT connection.
DAC2_FORCE_DATA_EN	1	0x0	0=DAC2 FORCE logic disabled 1=DAC2 input data forced as per DAC_FORCE_DATA_SEL and DAC_FORCE_DATA fields
DAC_FORCE_BLANK_OFF_EN	4	0x0	When doing DAC comparator test this is best set to 1 so horizontal and vertical retraces do not interrupt the testing. 0=Normal DAC BLANK functionality. 1=DAC BLANK forced off. Use this setting with DAC FORCE logic to detect CRT connection.
DAC_FORCE_DATA_EN	5	0x0	Enables DAC force logic. 0=DAC FORCE logic disabled 1=DAC input data forced as per DAC_FORCE_DATA_SEL and DAC_FORCE_DATA fields
DAC_FORCE_DATA_SEL	7:6	0x0	Selects combination of black (0x00) and DAC_FORCE_DATA value forced on red, green and blue DAC channels. 0=R=DAC_FORCE_DATA,G=B=0x00 1=G=DAC_FORCE_DATA,R=B=0x00 2=B=DAC_FORCE_DATA,R=G=0x00 3=R=G=B=DAC_FORCE_DATA
DAC_FORCE_DATA	17:8	0x0	Data forced on DAC channels as per DAC_FORCE_DATA_SEL and DAC_FORCE_DATA_EN.

DAC force control for analog monitor connection detection. This is used in combination with the DAC_CNTL register comparator fields.

DAC_CRC_SIG1 - R - 32 bits - [MMReg:0xD18]			
Field Name	**Bits**	**Default**	**Description**
DAC_CRC_SIG_B	9:0	0x0	CRC signature value for blue data. When reset, CRCB(9:0) <= 0x3FF; When CRC is running, CRCB(9:1) <= CRCB(8:0) xor blue(9:1), CRCB(0) <= CRCB(9) xor CRCB(2) xor blue(0)

DAC_CRC_SIG_G	25:16	0x0	CRC signature value for green data. When reset, CRCG(9:0) <= 0x3FF; When CRC is running, CRCG(9:1) <= CRCG(8:0) xor green(9:1), CRCG(0) <= CRCG(9) xor CRCG(2) xor green(0)

No description available for this register.

DAC_CRC_SIG2 - R - 32 bits - [MMReg:0xD1C]			
Field Name	**Bits**	**Default**	**Description**
DAC_CRC_SIG_R	9:0	0x0	CRC signature value for red data. When reset, CRCR(9:0) <= 0x3FF; When CRC is running, CRCR(9:1) <= CRCR(8:0) xor red(9:1), CRCR(0) <= CRCR(9) xor CRCR(2) xor red(0)
DAC_CRC_SIG_C	21:16	0x0	CRC signature value for control signals. The control signal input vector is made up as follows: control(0) <= not blank (0 in retrace, 1 in active or overscan) control(1) <= Display Enable (0 in retrace or overscan, 1 in active) control(2) <= HSYNC control(3)<=VSYNC control(4) <= AUXWIN control(5) <= STEREOSYNC When reset, CRCC(5:0) <= 0x3F; When CRC is running, CRCC(5:1) <= CRCC(4:0) xor control(5:1), CRCC(0) <= CRCC(5) xor CRCC(0) xor control(0)

CRC signature value for red component and control signals. Use DAC_CRC_EN to initate a field or frame analysis. See DAC_CRC_EN for details.

DAC_CRC2_SIG1 - R - 32 bits - [MMReg:0xD70]			
Field Name	**Bits**	**Default**	**Description**
DAC_CRC2_SIG_B	9:0	0x0	CRC signature value for blue data. When reset, CRCB(9:0) <= 0x3FF; When CRC is running, CRCB(9:1) <= CRCB(8:0) xor blue(9:1), CRCB(0) <= CRCB(9) xor CRCB(2) xor blue(0)

DAC_CRC2_SIG_G	25:16	0x0	CRC signature value for green data. When reset, CRCG(9:0) <= 0x3FF; When CRC is running, CRCG(9:1) <= CRCG(8:0) xor green(9:1), CRCG(0) <= CRCG(9) xor CRCG(2) xor green(0)

Secondary display path CRC signature vales for the blue and green components

DAC_CRC2_SIG2 - R - 32 bits - [MMReg:0xD74]			
Field Name	**Bits**	**Default**	**Description**
DAC_CRC2_SIG_R	9:0	0x0	CRC signature value for red data. When reset, CRCR(9:0) <= 0x3FF; When CRC is running, CRCR(9:1) <= CRCR(8:0) xor red(9:1), CRCR(0) <= CRCR(9) xor CRCR(2) xor red(0)
DAC_CRC2_SIG_C	21:16	0x0	CRC signature value for control signals. The control signal input vector is made up as follows: control(0) <= not blank (0 in retrace, 1 in active or overscan) control(1) <= Display Enable (0 in retrace or overscan, 1 in active) control(2) <= HSYNC control(3)<=VSYNC control(4) <= AUXWIN control(5) <= STEREOSYNC When reset, CRCC(5:0) <= 0x3F; When CRC is running, CRCC(5:1) <= CRCC(4:0) xor control(5:1), CRCC(0) <= CRCC(5) xor CRCC(0) xor control(0)

Secondary display path CRC signature value for red component and control signals. Use DAC_CRC_EN to initate a field or frame analysis. See DAC_CRC_EN for details.

DAC_CNTL2 - RW - 32 bits - [IOReg,MMReg:0x7C]			
Field Name	**Bits**	**Default**	**Description**
DAC_CLK_SEL (DISPLAY)	0	0x0	0=Selects Primary Display (CRTC1) as source for CRT DAC 1=Selects Secondary Display (CRTC2) as source for CRT DAC
DAC2_CLK_SEL (DISPLAY)	1	0x0	0=Selects TV OUT as source for TV DAC 1=Use TV DAC as secondary CRT. For source of display check CRT2_DISP1_SEL@DISP_HW_DEBUG

PALETTE_ACCESS_CNTL (DISPLAY)	5	0x0	0=Access Primary Palette through PALETTE_INDEX and PALETTE_DATA registers 1=Access Secondary Palette through PALETTE_INDEX and PALETTE_DATA registers
DAC2_CMP_EN (DISPLAY)	7	0x1	
DAC2_CMP_OUT_R (R) (DISPLAY)	8	0x0	
DAC2_CMP_OUT_G (R) (DISPLAY)	9	0x0	
DAC2_CMP_OUT_B (R) (DISPLAY)	10	0x0	
DAC2_CMP_OUTPUT (R) (DISPLAY)	11	0x0	
DAC2_EXPAND_MODE (DISPLAY)	14	0x0	0=Convert 4, 5 & 6 bit colors to 8 bits by zero extension 1=Convert 4, 5 & 6 bit colors to 8 bits by dynamic expansion
CRT2_SENSE (R) (DISPLAY)	16	0x0	0=Secondary CRT Monitor Detection result - not connected 1=Secondary CRT Monitor Detection result - connected
CRT2_DETECTION_ON (DISPLAY)	17	0x0	0=Secondary CRT Monitor Detection disabled 1=Secondary CRT Monitor Detection enabled
DAC_CRC2_CONT_EN (DISPLAY)	18	0x0	0=DAC CRC2 runs in one shot mode. 1=DAC CRC2 runs in continuous mode.
DAC_CRC2_EN (DISPLAY)	19	0x0	0=Reset DAC CRC2. 1=Enable DAC CRC2 for next frame/field in one shot, or all frames/fields in continuous mode.
DAC_CRC2_FIELD (DISPLAY)	20	0x0	0=If interlace display and one shot mode, then CRC even field only 1=If interlace display and one shot mode, then CRC odd field only
DAC2_LUT_COUNTER_LIMIT (DISPLAY)	22:21	0x0	0=anti-sparkle timeout 3 clocks 1=anti-sparkle timeout 7 clocks 2=anti-sparkle timeout 15 clocks 3=anti-sparkle timeout 31 clocks

No description available for this register.

DAC_MACRO_CNTL - RW - 32 bits - [MMReg:0xD04]			
Field Name	**Bits**	**Default**	**Description**
DAC_WHITE_CNTL	3:0	0x8	
DAC_BG_ADJ	11:8	0x8	
DAC_PDWN_R	16	0x0	
DAC_PDWN_G	17	0x0	
DAC_PDWN_B	18	0x0	

Graphics & cursor base address

2.17 vgaStat Registers

GENFC_RD - R - 8 bits - VGA_IO:0x3CA			
Field Name	**Bits**	**Default**	**Description**
VSYNC_SEL_R *(mirror of GENFC_WT:VSYNC_SEL_W)*	3	0x0	Veritcal sync select (read). 0=Normal vertical sync 1=Sync is 'vertical sync' ORed with 'vertical display enable'

Feature Control Regsiter (Read)

GENFC_WT - W - 8 bits - [VGA_IO:0x3BA] [VGA_IO:0x3DA]			
Field Name	**Bits**	**Default**	**Description**
VSYNC_SEL_W	3	0x0	Vertical sync select (write). 0=Normal vertical sync 1=Sync is 'vertical sync' ORed with 'vertical display enable'

Feature Control Register (Read)

GENMO_WT - W - 8 bits - VGA_IO:0x3C2			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B (BIF)	0	0x0	VGA addressing mode. 0=Monochrome emulation, regs at 0x3Bx 1=Color/Graphic emulation, regs at 0x3Dx
VGA_RAM_EN (BIF)	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture. 0=Disable 1=Enable
VGA_CKSEL (DISPLAY)	3:2	0x0	Selects pixel clock frequency to use in VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=0. See CLOCK_CNTL_INDEX.PPLL_DIV_SEL for non-VGA mode pixel clock selection. 0=25.1744MHz (640 Pels) 1=28.3212MHz (720 Pels) 2=Reserved 3=Reserved

ODD_EVEN_MD_PGSEL (HDP)	5	0x0	This bit is used in odd/even display modes (A/N modes: 0, 1, 2, 3, and 7). This bit is ignored when either bit GRA06[1] or SEQ4[3] are enabled. Used to determine if the VGA aperture maps into the lower (even) or upper (odd) page of memory. 0=Selects odd (high) memory locations 1=Selects even (low) memory locations
VGA_HSYNC_POL (DISPLAY)	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low The covention of VGA is to use active low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL (DISPLAY)	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low The covention of VGA is to use active high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

Miscellaneous Output Register (Write)

GENMO_RD - R - 8 bits - VGA_IO:0x3CC			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B *(mirror of GENMO_WT:GENMO_MONO_ADDRESS_B)* (BIF)	0	0x0	VGA addressing mode.
VGA_RAM_EN *(mirror of GENMO_WT:VGA_RAM_EN)* (BIF)	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture.
VGA_CKSEL *(mirror of GENMO_WT:VGA_CKSEL)* (DISPLAY)	3:2	0x0	Selects pixel clock frequency to use.

ODD_EVEN_MD_PGSEL *(mirror of GENMO_WT:ODD_EVEN_MD_PGSEL)* (HDP)	5	0x0	This bit is used in odd/even display modes (A/N modes: 0, 1, 2, 3, and 7). This bit is ignored when either bit GRA06[1] or SEQ4[3] are enabled. Used to determine if the VGA aperture maps into the lower (even) or upper (odd) page of memory.
VGA_HSYNC_POL *(mirror of GENMO_WT:VGA_HSYNC_POL)* (DISPLAY)	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low The covention of VGA is to use active low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL *(mirror of GENMO_WT:VGA_VSYNC_POL)* (DISPLAY)	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low The covention of VGA is to use active high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

Miscellaneous Output Register (Read)

GENS0 - R - 8 bits - VGA_IO:0x3C2			
Field Name	**Bits**	**Default**	**Description**
SENSE_SWITCH	4	0x0	DAC comparator read back. Used for monitor detection. Mirror of DAC_CMP_OUTPUT@DAC_CNTL. See description there.
CRT_INTR	7	0x0	CRT Interrupt: 0=Vertical retrace interrupt is cleared 1=Vertical retrace interrupt is pending

Feature Control Regsiter (Read)

GENS1 - R - 8 bits - [VGA_IO:0x3BA] [VGA_IO:0x3DA]			
Field Name	**Bits**	**Default**	**Description**
NO_DIPLAY	0	0x0	Display enable. 0=Enable 1=Disable
VGA_VSTATUS	3	0x0	Vertical Retrace Status. 0=Vertical retrace not active 1=Vertical retrace active

PIXEL_READ_BACK	5:4	0x0	Diagnostic bits 0, 1 respectively. These two bits are connected to two of the eight colour outputs (P7:P0) of the attribute controller. Connections are controlled by ATTR12(5,4) as follows: 0=P2,P0 1=P5,P4 2=P3,P1 3=P7,P6

Feature Control Regsiter (Read)

GRPH_BUFFER_CNTL - RW - 32 bits - [MMReg:0x2F0]			
Field Name	**Bits**	**Default**	**Description**
GRPH_START_REQ	6:0	0x5c	Request watermark where display requests to memory controller will start. Normally set to the upper limit. NOTE: Bits 0:1 of this field are hardwired to ZERO.
GRPH_STOP_REQ	14:8	0x5c	Request watermark where display requests to memory controller will stop. Normally set to the upper limit. NOTE: Bits 0:1 of this field are hardwired to ZERO.
GRPH_CRITICAL_POINT	22:16	0x20	Read data watermark where pending display requests to the memory controller go to high priority. This forces the memory controller to service the pending display requests ASAP. Setting 0x00 forces all display requests to remain high priority at all times.
GRPH_CRITICAL_CNTL	28	0x0	For debug. Should always be set low. 0=Critical point based on current lowest write point. 1=Critical point based on last write point.
GRPH_BUFFER_SIZE	29	0x1	For debug. Should always be set high. 0=Extended mode graphics buffer is 64 entries maximum. 1=Extended mode graphics buffer is 96 entries maximum.
GRPH_CRITICAL_AT_SOF	30	0x0	For debug. Should always be set low. 0=Wait 1/2 line before going critical at start of each frame. 1=Go critical immediately at the start of each frame.
GRPH_STOP_CNTL	31	0x0	For debug. Should always be set low. 0=GRPH_STOP_REQ and VGA_STOP_REQ are -4 from actual stop point. 1=GRPH_STOP_REQ and VGA_STOP_REQ are extact stop point.

Control of display buffer fill requests for non-VGA modes. This register is programmed by the BIOS or DAL on display mode switches.

VGA_BUFFER_CNTL - RW - 32 bits - [MMReg:0x2F4]			
Field Name	**Bits**	**Default**	**Description**
VGA_START_REQ	5:0	0x3c	Request watermark where display requests to memory controller will start. NOTE: Bits 0:1 of this field are hardwired to ZERO.
VGA_STOP_REQ	13:8	0x3c	Request watermark where display requests to memory controller will stop. NOTE: Bits 0:1 of this field are hardwired to ZERO.
VGA_CRITICAL_POINT	21:16	0x20	Read data watermark where pending display requests to the memory controller go to high priority. Setting 0x00 forces all display requests to remain high priority at all times.

Control of display buffer fill requests for VGA modes. This register is programmed by the BIOS on VGA mode switches.

GRPH2_BUFFER_CNTL - RW - 32 bits - [MMReg:0x3F0]			
Field Name	**Bits**	**Default**	**Description**
GRPH2_START_REQ	6:0	0x5c	Request watermark where display requests to memory controller will start. Normally set to the upper limit. NOTE: Bits 0:1 of this field are hardwired to ZERO.
GRPH2_STOP_REQ	14:8	0x5c	Request watermark where display requests to memory controller will stop. Normally set to the upper limit. NOTE: Bits 0:1 of this field are hardwired to ZERO.
GRPH2_CRITICAL_POINT	22:16	0x20	Read data watermark where pending display requests to the memory controller go to high priority. This forces the memory controller to service the pending display requests ASAP. Setting 0x00 forces all display requests to remain high priority at all times.
GRPH2_CRITICAL_CNTL	28	0x0	For debug. Should always be set low. 0=Critical point based on current lowest write point. 1=Critical point based on last write point.
GRPH2_BUFFER_SIZE	29	0x1	For debug. Should always be set high. 0=Extended mode graphics buffer is 64 entries maximum. 1=Extended mode graphics buffer is 96 entries maximum.
GRPH2_CRITICAL_AT_SOF	30	0x0	For debug. Should always be set low. 0=Wait 1/2 line before going critical at start of each frame. 1=Go critical immediately at the start of each frame.

GRPH2_STOP_CNTL	31	0x0	For debug. Should always be set low. 0=GRPH_STOP_REQ is -4 from actual stop point. 1=GRPH_STOP_REQ is extact stop point.

Control of secondary display buffer fill requests. This register is programmed by the BIOS or DAL on display mode switches.

GEN_INT_CNTL - RW - 32 bits - [IOReg,MMReg:0x40]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_MASK (DISPLAY)	0	0x0	Vertical blank interrupt mask. 0=Disable 1=Enable
CRTC_VLINE_MASK (DISPLAY)	1	0x0	Vertical line interrupt mask. 0=Disable 1=Enable
CRTC_VSYNC_MASK (DISPLAY)	2	0x0	Vertical sync interrupt mask. 0=Disable 1=Enable
SNAPSHOT_MASK (DISPLAY)	3	0x0	Snapshot interrupt mask. 0=Disable 1=Enable
FP_DETECT_MASK (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt mask. 0=Disable 1=Enable
CRTC2_VLINE_MASK (DISPLAY)	5	0x0	0=Disable 1=Enable
CRTC2_VSYNC_MASK (DISPLAY)	6	0x0	0=Disable 1=Enable
SNAPSHOT2_MASK (DISPLAY)	7	0x0	0=Disable 1=Enable
CRTC2_VBLANK_MASK (DISPLAY)	9	0x0	0=Disable 1=Enable
FP2_DETECT_MASK (DISPLAY)	10	0x0	0=Disable 1=Enable
VSYNC_DIFF_OVER_LIMIT_MASK (TVOUT)	11	0x0	0=Disable 1=Enable

GUI_IDLE_MASK (RBBM)	19	0x0	GUI idle interrupt mask. 0=Disable 1=Enable
SW_INT_EN (HDP)	25	0x0	Software interrupt mask. 0=Disable 1=Enable
GEYSERVILLE_MASK (VIP)	27	0x0	0=Disable 1=Enable
HDCP_AUTHORIZED_INT_MASK (DISPLAY)	28	0x0	0=Disable 1=Enable
DVI_I2C_INT_MASK (DISPLAY)	29	0x0	0=Disable 1=Enable
GUIDMA_MASK (CP)	30	0x0	GUI DMA channel interrupt mask. 0=Disable 1=Enable
VIDDMA_MASK (CP)	31	0x0	Video capture DMA channel interrupt mask. 0=Disable 1=Enable

General Interrupt Control register.
The MASK/EN fields control whether the respective status bits are enabled to drive the system interrupt pin. Even if enabled here, the interrupt line is not driven unless enabled in the PCI configuration space.

GEN_INT_STATUS - RW - 32 bits - [IOReg,MMReg:0x44]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_STAT (R) (DISPLAY)	0	0x0	Vertical blank interrupt. Set when display in vertical retrace. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VBLANK_STAT_AK (W) (DISPLAY)	0	0x0	Vertical blank interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VLINE_STAT (R) (DISPLAY)	1	0x0	Vertical line interrupt. Set on display line on programmed by the CRTC_VLINE_CRNT_VLINE.CRTC_VLINE register. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VLINE_STAT_AK (W) (DISPLAY)	1	0x0	Vertical line interrupt acknowledge. 0=No effect 1=Clear status

CRTC_VSYNC_STAT (R) (DISPLAY)	2	0x0	Vertical sync interrupt. Set on start of VSYNC at the DAC. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_STAT_AK (W) (DISPLAY)	2	0x0	Vertical sync interrupt acknowledge. 0=No effect 1=Clear status
SNAPSHOT_STAT (R) (DISPLAY)	3	0x0	Snapshot interrupt. Set as controlled by SNAPSHOT_VIF_COUNT register. 0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT_STAT_AK (W) (DISPLAY)	3	0x0	Snapshot interrupt acknowledge. 0=No effect 1=Clear status
FP_DETECT_STAT (R) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt. Set on HPD connect or disconnect as controlled by FP_GEN_CNTL.FP_DETECT_INT_POL. 0=No event 1=Event has occurred, interrupting if enabled
FP_DETECT_STAT_AK (W) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt acknowledge. 0=No effect 1=Clear status
CRTC2_VLINE_STAT (R) (DISPLAY)	5	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VLINE_STAT_AK (W) (DISPLAY)	5	0x0	0=No effect 1=Clear status
CRTC2_VSYNC_STAT (R) (DISPLAY)	6	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VSYNC_STAT_AK (W) (DISPLAY)	6	0x0	0=No effect 1=Clear status
SNAPSHOT2_STAT (R) (DISPLAY)	7	0x0	0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT2_STAT_AK (W) (DISPLAY)	7	0x0	0=No effect 1=Clear status
CAP0_INT_ACTIVE (R) (VIP)	8	0x0	Capture port 0 has active interrupt(s). 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CRTC2_VBLANK_STAT (R) (DISPLAY)	9	0x0	0=No event 1=Event has occurred, interrupting if enabled

CRTC2_VBLANK_STAT_AK (W) (DISPLAY)	9	0x0	0=No effect 1=Clear status
FP2_DETECT_STAT (R) (DISPLAY)	10	0x0	0=No event 1=Event has occurred, interrupting if enabled
FP2_DETECT_STAT_AK (W) (DISPLAY)	10	0x0	0=No effect 1=Clear status
VSYNC_DIFF_OVER_LIMIT_STAT (R) (TVOUT)	11	0x0	0=No event 1=Event has occurred, interrupting if enabled
VSYNC_DIFF_OVER_LIMIT_STAT_AK (W) (TVOUT)	11	0x0	0=No effect 1=Clear status
GUI_IDLE_STAT (R) (RBBM)	19	0x1	GUI idle interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_STAT_AK (W) (RBBM)	19	0x0	GUI idle interrupt acknowledge. 0=No effect 1=Clear status
SW_INT (R) (HDP)	25	0x0	Software interrupt. General purpose interrupt that can only be set by software event by writing to SW_INT_SET. 0=No event 1=Event has occurred, interrupting if enabled
SW_INT_AK (W) (HDP)	25	0x0	Software interrupt acknowledge. 0=No effect 1=Clear SW_INT (set low)
SW_INT_SET (W) (HDP)	26	0x0	Software interrupt trigger. 0=No effect 1=Set SW_INT active (high)
GEYSERVILLE_STAT (R) (VIP)	27	0x0	0=No event 1=Event has occurred, interrupting if enabled
GEYSERVILLE_STAT_AK (W) (VIP)	27	0x0	0=No effect 1=Clear status
HDCP_AUTHORIZED_INT_STAT (R) (DISPLAY)	28	0x0	0=No event 1=Event has occurred, interrupting if enabled
HDCP_AUTHORIZED_INT_AK (W) (DISPLAY)	28	0x0	0=No effect 1=Clear status

DVI_I2C_INT_STAT (R) (DISPLAY)	29	0x0	0=No event 1=Event has occurred, interrupting if enabled
DVI_I2C_INT_AK (W) (DISPLAY)	29	0x0	0=No effect 1=Clear status
GUIDMA_STAT (R) (CP)	30	0x0	GUI DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUIDMA_AK (W) (CP)	30	0x0	GUI DMA channel interrupt acknowledge. 0=No effect 1=Clear status
VIDDMA_STAT (R) (CP)	31	0x0	Video capture DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
VIDDMA_AK (W) (CP)	31	0x0	Video capture DMA channel interrupt acknowledge. 0=No effect 1=Clear status

General Interrupt Status register.
These fields can be polled and acknowledged even if interrupts are disabled, or the respective fields are masked in the GEN_INT_CNTL register.

2.18 vgaSeq Registers

SEQ00 - RW - 8 bits - VGASEQIND:0x0			
Field Name	**Bits**	**Default**	**Description**
SEQ_RST0B	0	0x1	Synchronous reset bit 0: 0=Follows SEQ_RST1B 1=Sequencer runs unless SEQ_RST1B=0
SEQ_RST1B	1	0x1	Synchronous reset bit 1: 0=Disable character clock, display requests, and H/V syncs 1=Sequencer runs unless SEQ_RST0B=0

Reset Register

SEQ01 - RW - 8 bits - VGASEQIND:0x1			
Field Name	**Bits**	**Default**	**Description**
SEQ_DOT8	0	0x1	8/9 Dot Clocks (Modes 1, 2, 3, and 7 use 9-dot characters. To change bit 0, GENVS(0) must be logical 0). 0=9 dot char clock. Modes 0, 1, 2, 3 & 7 1=8 dot char clock.
SEQ_SHIFT2	2	0x0	Shift load bits. 0=Load video serializer every clock, if SEQ_SHIFT4=0 1=Load video serializer every other clock, if SEQ_SHIFT4=0
SEQ_PCLKBY2	3	0x0	Dot Clock (typically, 320 and 360 horizontal modes use divide-by-2 to provide 40 column displays. To change this bit SEQ00[0:0] must be first set to zero.)). 0=Dot clock is normal 1=Dot clock is divided by 2
SEQ_SHIFT4	4	0x0	Shift load bits. 0=SEQ_SHIFT2 determines serializer loading 1=Load video serializer every fourth clock. Ignore SEQ_SHIFT2
SEQ_MAXBW	5	0x1	Screen off: 0=Normal. Screen on 1=Sreen off and blanked. CPU has uninterrupted access to frame buffer

Clock Mode Register

SEQ03 - RW - 8 bits - VGASEQIND:0x3			
Field Name	**Bits**	**Default**	**Description**
SEQ_FONT_B1	0	0x0	Character Map Select B Bit 1
SEQ_FONT_B2	1	0x0	Character Map Select B Bit 2
SEQ_FONT_A1	2	0x0	Character Map Select A Bit 1
SEQ_FONT_A2	3	0x0	Character Map Select A Bit 2
SEQ_FONT_B0	4	0x0	Character Map Select B Bit 0
SEQ_FONT_A0	5	0x0	Character Map Select A Bit 0

Character Map Select Register

SEQ8_IDX - RW - 8 bits - VGA_IO:0x3C4			
Field Name	**Bits**	**Default**	**Description**
SEQ_IDX (HDP) (DISPLAY)	2:0	0x0	This index points to one of the sequencer registers (SEQ_) at I/O port address 0x3C5, for the next SEQ read/write operation.

SEQ Index Register

SEQ8_DATA - RW - 8 bits - VGA_IO:0x3C5			
Field Name	**Bits**	**Default**	**Description**
SEQ_DATA (HDP) (DISPLAY)	7:0	0x0	SEQ data indirect access

SEQ Data Register

2.19 vgaCrt Registers

CRTC8_IDX - RW - 8 bits - [VGA_IO:0x3B4] [VGA_IO:0x3D4]			
Field Name	**Bits**	**Default**	**Description**
VCRTC_IDX *(mirror bits 0:5 of CRTC_EXT_CNTL:VCRTC_IDX_MASTER)* (HDP) (DISPLAY)	5:0	0x0	This index points to one of the internal registers of the CRT controller (CRTC) at address 0x3?5, for the next CRTC read/write operation.

CRT Index Register

CRTC8_DATA - RW - 8 bits - [VGA_IO:0x3B5] [VGA_IO:0x3D5]			
Field Name	**Bits**	**Default**	**Description**
VCRTC_DATA (HDP) (DISPLAY)	7:0	0x0	CRTC data indirect access

CTRC Data Register

CRT00 - RW - 8 bits - VGACRTIND:0x0			
Field Name	**Bits**	**Default**	**Description**
H_TOTAL *(mirror bits 0:7 of CRTC_H_TOTAL_DISP:CRTC_H_TOTAL)*	7:0	0x0	These bits define the active horizontal display in a scan line, including the retrace period. The value is five less than the total number of displayed characters in a scan line.

Horizontal Total Register

CRT01 - RW - 8 bits - VGACRTIND:0x1			
Field Name	**Bits**	**Default**	**Description**
H_DISP_END *(mirror bits 0:7 of CRTC_H_TOTAL_DISP:CRTC_H_DISP)*	7:0	0x0	These bits define the active horizontal dispaly in a scan line. The value is one less than the total number of displayed characters in a scan line.

Horizontal Display Enable End Register

CRT02 - RW - 8 bits - VGACRTIND:0x2			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_START	7:0	0x0	These bits define the horizontal character count that represents the character coune in the active display area plus the right borger. In other words, the count is from the start of active display to the start of triggering of the H blanking pulse.

Start Horizontal Blanking Register

CRT03 - RW - 8 bits - VGACRTIND:0x3			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_END	4:0	0x0	H blanking bits 4-0 respectively. These are the five low-order bits (of six bits in total) of horizontal character count for triggering the end of the horizontal blanking pulse.
H_DE_SKEW	6:5	0x0	Display-enable skew: 0=0Skew 1=1Skew 2=2Skew 3=3Skew
CR10CR11_R_DIS_B	7	0x0	Comptibility Read: 0=WrtOnlyToCRT10-11 1=WrtRdToCRT10-11

End Horizontal Blanking Register

CRT04 - RW - 8 bits - VGACRTIND:0x4			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_START *(mirror bits 0:7 of CRTC_H_SYNC_STRT_WID:CRTC_H_SYNC_STRT_CHAR)*	7:0	0x0	These bits define the horizontal character count at which the horizontal retrace pulse becomes active.

Start Horizontal Retrace Register

CRT05 - RW - 8 bits - VGACRTIND:0x5			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_END *(mirror bits 0:4 of CRTC_H_SYNC_STRT_WID:CRTC_H_SYNC_WID)*	4:0	0x0	H Retrace Bits (these are the 5-bit result from the sum of CRT0 plus the width of the horizontal retrace pulse, in character clock units).
H_SYNC_SKEW	6:5	0x0	H Retrace Delay bits (these two bits skew the horizontal retrace pulse).
H_BLANK_END_B5	7	0x0	H blocking end bit 5 (this is the bit of the 6-bit character count for the H blanking end pulse). The other five low-order bits are CRT03[4:0].

End Horizontal Retrace Register

CRT06 - RW - 8 bits - VGACRTIND:0x6			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL *(mirror bits 0:7 of CRTC_V_TOTAL_DISP:CRTC_V_TOTAL)*	7:0	0x0	These are the eight low-order bits of the 10-bit vertical total register. The 2 high-order bits are CRT07[5:0] in the CRTC overflow register. The value of this register represents the total number of H raster scans plus vertical retrace (active display, blanking), minus two scan lines.

Vertical Total Register

CRT07 - RW - 8 bits - VGACRTIND:0x7			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL_B8 *(mirror bits 8:8 of CRTC_V_TOTAL_DISP:CRTC_V_TOTAL)*	0	0x0	V Total Bit 8 (CRT06). Bit 8 of 10 bit vertical count for V Total. For functional description see CRT06 register.
V_DISP_END_B8 *(mirror bits 8:8 of CRTC_V_TOTAL_DISP:CRTC_V_DISP)*	1	0x0	End V Display Bit 8 (CRT12). Bit 8 of 10-bit vertical count for V Display enable. For functional desription see CRT12 register.
V_SYNC_START_B8 *(mirror bits 8:8 of CRTC_V_SYNC_STRT_WID:CRTC_V_SYNC_STRT)*	2	0x0	Start V Retrace Bit 8 (CRT10). Bit 8 of 10-bit veritcal count for V Retrace start. For functional description see CRT10 register.

V_BLANK_START_B8	3	0x0	Start V Blanking Bit 8 (CRT15). Bit 8 of the 10-bit vertical count for V Blanking start. For functional description see CRT15 register.
LINE_CMP_B8 *(mirror bits 8:8 of CRTC_VLINE_CRNT_VLINE:CRTC_VLINE)*	4	0x0	Line compare bit 8 (CRT18). Bit 8 of the 10-bit vertical count for line compare. For functional description see CRT18 register.
V_TOTAL_B9 *(mirror bits 9:9 of CRTC_V_TOTAL_DISP:CRTC_V_TOTAL)*	5	0x0	V Total Bit 9 (CRT06). Bit 9 of 10-bit vertical count for V Total. For functional description see CRT06 register.
V_DISP_END_B9 *(mirror bits 9:9 of CRTC_V_TOTAL_DISP:CRTC_V_DISP)*	6	0x0	End V Display Bit 9 (CRT12). Bit 9 of 10-bit vertical count for V Display enable end (for functional description see CRT12 register).
V_SYNC_START_B9 *(mirror bits 9:9 of CRTC_V_SYNC_STRT_WID:CRTC_V_SYNC_STRT)*	7	0x0	Start V Retrace Bit (CRT10). Bit 9 of 10-bit vertical count for V Retrace start. For functional description see CRT10 register.

CRTC Overflow Register

CRT08 - RW - 8 bits - VGACRTIND:0x8			
Field Name	**Bits**	**Default**	**Description**
ROW_SCAN_START	4:0	0x0	Preset row scan bit 4:0. This register is used for software-controlled vertical scrolling in text or graphics modes. The value specifies the first line to be scanned after a V retrace (in the next frame). Each H Retrace pulse increments the counter by 1, up to the maximum scan line value programmed by CRT09, then the counter is cleared.
BYTE_PAN	6:5	0x0	Byte panning control bits 1 and 0 (respectively). Bits 6 and 5 extend the capability of byte panning (shifting) by up to three characters (for description H_PEL Panning register ATTR13).

Preset Row Scan Register

CRT09 - RW - 8 bits - VGACRTIND:0x9			
Field Name	**Bits**	**Default**	**Description**
MAX_ROW_SCAN	4:0	0x0	Maximum scan line bits. These bits define a value that is the actual number of scan line per character minus 1.

V_BLANK_START_B9	5	0x0	Start V Blanking bit 9 (CRT15). Bit 9 of 10-bit veritcal count for line compare. For functional description see CRT18 register.
LINE_CMP_B9 *(mirror bits 9:9 of CRTC_VLINE_CRNT_VLINE:CRTC_VLINE)*	6	0x0	Line Compare Bit 9 (CRT18). Bit 9 of 10-bit vertical count for line compare. For functional description see CRT18 register.
DOUBLE_CHAR_HEIGHT	7	0x0	200/400 line scan. NOTE H/V display and blanking timings etc. (in CRT00-CRT06 registers) are not affected. 0=200LineScan 1=400LineScan

Maximum Scan Line Register

CRT0A - RW - 8 bits - VGACRTIND:0xA			
Field Name	**Bits**	**Default**	**Description**
CURSOR_START	4:0	0x0	Cursor start bits 4:0 (respectively). These bits define a value that is the starting scan line (on a character row) for the line cursor. The 5-bit value is equal to the actual number minus one. This value is used together with the Cursor End Bits CRT0B[4:0] to determine the height of the cursor. The cursor height in VGA does not wrap around (as in EGA) and is actually absent when the 'end' value is less than the 'start' value. In EGA when the 'end' value is less, the cursor is a full block cursor the same height as the character cell.
CURSOR_DISABLE	5	0x0	Cursor on/off. 0=on 1=off

Cursor Start Register

CRT0B - RW - 8 bits - VGACRTIND:0xB			
Field Name	**Bits**	**Default**	**Description**
CURSOR_END	4:0	0x0	Cursor End Bits 4-0, respectively.- These bits define the ending scan row (on a character line) for the line cursor. In EGA, this 5-bit value is equal to the actual number of lines plus one.- The cursor height in VGA does not wrap around (as in EGA) and is actually absent when the 'end' value is less than the 'start' value. In EGA when the 'end' value is less, the cursor is a full block cursor the same height as the character cell.

CURSOR_SKEW	6:5	0x0	Cursor Skew Bits 1 and 0, respectively.- These bits define the number of characters the cursor is to be shifted to the right (skewed) from the character pointed at by the cursor location (registers CRT0E and CRT0F), in VGA mode. Skew values when in EGA mode are enclosed in brackets.

Cursor End Register

CRT0C - RW - 8 bits - VGACRTIND:0xC			
Field Name	**Bits**	**Default**	**Description**
DISP_START *(mirror bits 11:18 of CRTC_OFFSET:CRTC_OFFSET)*	7:0	0x0	SA bits 15:8-These are the eight high-order bits of the 16-bit display buffer start location. The low order bits are contained in CRT0D.-In split screen mode, CRT0C = CRT0D point to the starting location of screen A (top half.) The starting address for screen B is always zero.

Start Address (High Byte) Register

CRT0D - RW - 8 bits - VGACRTIND:0xD			
Field Name	**Bits**	**Default**	**Description**
DISP_START *(mirror bits 3:10 of CRTC_OFFSET:CRTC_OFFSET)*	7:0	0x0	SA bits 7:0- These are the eight low-order bits of the 16-bit display buffer start location. The high-order bits are contained in CRT0C. - In split creen mode, CRT0C + CRT0D points to the starting location of screen A (top half.) The starting address for screen B is always zero.

Start Address (Low Byte) Register

CRT0E - RW - 8 bits - VGACRTIND:0xE			
Field Name	**Bits**	**Default**	**Description**
CURSOR_LOC_HI	7:0	0x0	CA bits 15:8- These are the eight high-order bits of the 16 bit cursor start address. The low-order CA bits are contained in CRT0F. This address is relative to the start of physical display memory address pointed to by CRT0C + CRT0D. In other words, if CRT0C + CRT0D is changed, the cursor still pints to the same character as before.

Cursor Location (High Byte) Register

CRT0F - RW - 8 bits - VGACRTIND:0xF			
Field Name	**Bits**	**Default**	**Description**
CURSOR_LOC_LO	7:0	0x0	CA bits 7:0- These are the eight low-order bits of the 16 bit cursor start address. The high-order CA bits are contained in CRT0E. This address is relative to the start of physical display memory address pointed to by CRT0C + CRT0D. In other words, if CRT0C + T0D is changed, the cursor still points to the same character as before

Cursor Location (Low Byte) Register

CRT10 - RW - 8 bits - VGACRTIND:0x10			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_START *(mirror bits 0:7 of CRTC_V_SYNC_STRT_WID:CRTC_V_SYNC_STRT)*	7:0	0x0	Bits CRT10[7:0] are the eight low-order bits of the 10-bit vertical retrace start count. The two high-order bits are CRTt07[2:7], located in the CRTC overflow register.- These bits define the horizontal scan count that triggers the V retrace pulse.

Start Vertical Retrace Register

CRT11 - RW - 8 bits - VGACRTIND:0x11			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_END *(mirror bits 0:3 of CRTC_V_SYNC_STRT_WID:CRTC_V_SYNC_WID)*	3:0	0x0	V Retrace End Bits 3-0- Bits CRT11[0:3] define the horizontal scan count that triggers the end of the V Retrace pulse.
V_INTR_CLR	4	0x0	V Retrace Interrupt Set: 0=VRetraceIntCleared 1=Not Cleared
V_INTR_EN	5	0x0	V Retrace Interrupt Disabled: 0=VRetraceIntEna 1=Disable
SEL5_REFRESH_CYC	6	0x0	0=3 DRAM Refresh/Horz Line 1=5 DRAM Refresh/Horz Line
C0T7_WR_ONLY	7	0x0	Write Protect (CRT00-CRT06). All register bits except CRTO7[4] are write protected. 0=EnaWrtToCRT00-07 1=C0T7B4WrtOnly

End Vertical Retrace Register

CRT12 - RW - 8 bits - VGACRTIND:0x12			
Field Name	**Bits**	**Default**	**Description**
V_DISP_END *(mirror bits 0:7 of CRTC_V_TOTAL_DISP:CRTC_V_DISP)*	7:0	0x0	These are the eight low-order bits of the 10-bit register containing the horizontal scan count indicating where the active display on the screen should end. The high-order bits are CRT07 [1:6] in the CRT overflow register.

Vertical Display Enable End Register

CRT13 - RW - 8 bits - VGACRTIND:0x13			
Field Name	**Bits**	**Default**	**Description**
DISP_PITCH *(mirror bits 0:7 of CRTC_PITCH:CRTC_PITCH)*	7:0	0x0	- These bits define an offset value, equal to the logical line width of the screen (from the first character of the current line to the first character of the next line).- Memory organization is dependent on the video mode. Bit CRT17[6] selects byte or word mode. Bit CRT14[6], which overrides the byte/word mode setting, selects Double-Word mode when it is logical one.- The first character of the next line is specified by the start address (CRT0C + CRT0D) plus the offset. The offset for byte mode is 2x CRT13; for word mode, 4x; for double word mode 8x.

Offset Register

CRT14 - RW - 8 bits - VGACRTIND:0x14			
Field Name	**Bits**	**Default**	**Description**
UNDRLN_LOC (DISPLAY)	4:0	0x0	H Row Scan Bits 4-0.- These bits define the horizontal scan row, from the top of the characterline, that should be used for underlining. The 5-bit value is equal to the actual number minus one.
ADDR_CNT_BY4 (DISPLAY)	5	0x0	Count-by-4: 0=Char. Clock 1=CountBy4
DOUBLE_WORD (HDP) (DISPLAY)	6	0x0	Double-Word Mode: 0=Disable 1=DoubleWordMdEna

Underline Location Register

CRT15 - RW - 8 bits - VGACRTIND:0x15			
Field Name	**Bits**	**Default**	**Description**
V_BLANK_START	7:0	0x0	These are the eight low-order bits of the 10-bit vertical blanking start register. Bit 9 is CRT09[5]; bit 8 is CRT07[3]- The 10 bits specify the starting location of the vertical blaning pulse, in units of horizontal scan lines. The value is equal to the actual number of displayed lines minus one.

Start Vertical Blanking Register

CRT16 - RW - 8 bits - VGACRTIND:0x16			
Field Name	**Bits**	**Default**	**Description**
V_BLANK_END	7:0	0x0	These bits define the point at which to trigger the end of the vertical blanking pulse. The location is specified in units of horizontal scan lines.- The value to be storeed in this register is the seven low-order bits of the sum of 'pulse width count' plus the content of Start Vertical Blanking register (CRT15) minus one.

End Vertical Blanking Register

CRT17 - RW - 8 bits - VGACRTIND:0x17			
Field Name	**Bits**	**Default**	**Description**
RA0_AS_A13B (DISPLAY)	0	0x0	Compatibility Mode:
RA1_AS_A14B (DISPLAY)	1	0x0	Select Row Scan Counter:
VCOUNT_BY2 (DISPLAY)	2	0x0	Vertical_by_2 NOTE: When bit 2 is logical one, other vertical register values shoudl be adjusted as well (CRT06, CRT10, CRT12, CRT15, and CRT18).
ADDR_CNT_BY2 (DISPLAY)	3	0x0	Count_by_2: ENGINEERING NOTE: Bit can be written and read, but has no effect.
WRAP_A15TOA0 (DISPLAY)	5	0x0	Address Wrap: ENGINEERING NOTE: Bit can be written and read, but has no effect.

BYTE_MODE (HDP) (DISPLAY)	6	0x0	Byte/Word Mode: 0=WordMode 1=ByteMode
CRTC_SYNC_EN (DISPLAY)	7	0x0	H/V Retrace Enable: 0=Disable HVSync 1=EnaHVSync

CRT Mode Register

CRT18 - RW - 8 bits - VGACRTIND:0x18			
Field Name	**Bits**	**Default**	**Description**
LINE_CMP *(mirror bits 0:7 of CRTC_VLINE_CRNT_VLINE:CRTC_VLINE)*	7:0	0x0	- These bits are the eight low-order of the 10-bit line compare register. Bit 8 is CRT07[4], bit 9 is CRT09[6]. The value of this register is used to disable scrolling on a portion of the display screen, as when split screen is active. When the vertical counter reaches this value, the memory address and row scan counters are cleared.- The screen area above the line specified by the register is commonly called screen A. The screen below is screen B. Screen B cannot be scrolled, but it can panned only together with screen A, controlled by the PEL panning compatibility bit ATTR10[5]. (For a description of this control bit see ATTR10[5].)

Line Compare Register

CRT00_S - RW - 8 bits - VGACRTIND:0x40			
Field Name	**Bits**	**Default**	**Description**
H_TOTAL_S *(mirror bits 0:7 of FP_CRTC_H_TOTAL_DISP:FP_CRTC_H_TOTAL)*	7:0	0x0	

No description available for this register.

CRT01_S - RW - 8 bits - VGACRTIND:0x41			
Field Name	**Bits**	**Default**	**Description**
H_DISP_END_S *(mirror bits 0:7 of FP_CRTC_H_TOTAL_DISP:FP_CRTC_H_DISP)*	7:0	0x0	

No description available for this register.

CRT02_S - RW - 8 bits - VGACRTIND:0x42			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_START_S	7:0	0x0	

No description available for this register.

CRT03_S - RW - 8 bits - VGACRTIND:0x43			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_END_S	4:0	0x0	
H_DE_SKEW_S	6:5	0x0	0=0Skew 1=1Skew 2=2Skew 3=3Skew
CR10CR11_R_DIS_B_M *(mirror of CRT03:CR10CR11_R_DIS_B)*	7	0x0	0=WrtOnlyToCRT10-11 1=WrtRdToCRT10-11

No description available for this register.

CRT04_S - RW - 8 bits - VGACRTIND:0x44			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_START_S *(mirror bits 0:7 of FP_H_SYNC_STRT_WID:FP_H_SYNC_STRT_CHAR)*	7:0	0x0	

No description available for this register.

CRT05_S - RW - 8 bits - VGACRTIND:0x45			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_END_S *(mirror bits 0:4 of FP_H_SYNC_STRT_WID:FP_H_SYNC_WID)*	4:0	0x0	
H_SYNC_SKEW_S	6:5	0x0	
H_BLANK_END_B5_S	7	0x0	

No description available for this register.

CRT06_S - RW - 8 bits - VGACRTIND:0x46			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL_S *(mirror bits 0:7 of FP_CRTC_V_TOTAL_DISP:FP_CRTC_V_TOTAL)*	7:0	0x0	

No description available for this register.

CRT07_S - RW - 8 bits - VGACRTIND:0x47			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL_B8_S *(mirror bits 8:8 of FP_CRTC_V_TOTAL_DISP:FP_CRTC_V_TOTAL)*	0	0x0	
V_DISP_END_B8_S *(mirror bits 8:8 of FP_CRTC_V_TOTAL_DISP:FP_CRTC_V_DISP)*	1	0x0	
V_SYNC_START_B8_S *(mirror bits 8:8 of FP_V_SYNC_STRT_WID:FP_V_SYNC_STRT)*	2	0x0	
V_BLANK_START_B8_S	3	0x0	

LINE_CMP_B8_M *(mirror bits 8:8 of CRTC_VLINE_CRNT_VLINE:CRTC_VLINE)*	4	0x0	
V_TOTAL_B9_S *(mirror bits 9:9 of FP_CRTC_V_TOTAL_DISP:FP_CRTC_V_TOTAL)*	5	0x0	
V_DISP_END_B9_S *(mirror bits 9:9 of FP_CRTC_V_TOTAL_DISP:FP_CRTC_V_DISP)*	6	0x0	
V_SYNC_START_B9_S *(mirror bits 9:9 of FP_V_SYNC_STRT_WID:FP_V_SYNC_STRT)*	7	0x0	

No description available for this register.

CRT08_S - RW - 8 bits - VGACRTIND:0x48			
Field Name	**Bits**	**Default**	**Description**
ROW_SCAN_START_M *(mirror of CRT08:ROW_SCAN_START)*	4:0	0x0	
BYTE_PAN_M *(mirror of CRT08:BYTE_PAN)*	6:5	0x0	

No description available for this register.

CRT09_S - RW - 8 bits - VGACRTIND:0x49			
Field Name	**Bits**	**Default**	**Description**
MAX_ROW_SCAN_S	4:0	0x0	
V_BLANK_START_B9_S	5	0x0	
LINE_CMP_B9_M *(mirror bits 9:9 of CRTC_VLINE_CRNT_VLINE:CRTC_VLINE)*	6	0x0	

DOUBLE_CHAR_HEIGHT_M *(mirror of CRT09:DOUBLE_CHAR_HEIGHT)*	7	0x0	0=200LineScan 1=400LineScan

No description available for this register.

CRT0A_S - RW - 8 bits - VGACRTIND:0x4A			
Field Name	**Bits**	**Default**	**Description**
CURSOR_START_S	4:0	0x0	
CURSOR_DISABLE_M *(mirror of CRT0A:CURSOR_DISABLE)*	5	0x0	0=on 1=off

No description available for this register.

CRT0B_S - RW - 8 bits - VGACRTIND:0x4B			
Field Name	**Bits**	**Default**	**Description**
CURSOR_END_S	4:0	0x0	
CURSOR_SKEW_M *(mirror of CRT0B:CURSOR_SKEW)*	6:5	0x0	

No description available for this register.

CRT0C_S - RW - 8 bits - VGACRTIND:0x4C			
Field Name	**Bits**	**Default**	**Description**
DISP_START_M *(mirror bits 11:18 of CRTC_OFFSET:CRTC_OFFSET)*	7:0	0x0	

No description available for this register.

CRT0D_S - RW - 8 bits - VGACRTIND:0x4D			
Field Name	**Bits**	**Default**	**Description**
DISP_START_M *(mirror bits 3:10 of CRTC_OFFSET:CRTC_OFFSET)*	7:0	0x0	

No description available for this register.

CRT0E_S - RW - 8 bits - VGACRTIND:0x4E			
Field Name	**Bits**	**Default**	**Description**
CURSOR_LOC_HI_M *(mirror of CRT0E:CURSOR_LOC_HI)*	7:0	0x0	

No description available for this register.

CRT0F_S - RW - 8 bits - VGACRTIND:0x4F			
Field Name	**Bits**	**Default**	**Description**
CURSOR_LOC_LO_M *(mirror of CRT0F:CURSOR_LOC_LO)*	7:0	0x0	

No description available for this register.

CRT10_S - RW - 8 bits - VGACRTIND:0x50			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_START_S *(mirror bits 0:7 of FP_V_SYNC_STRT_WID:FP_V_SYNC_STRT)*	7:0	0x0	

No description available for this register.

CRT11_S - RW - 8 bits - VGACRTIND:0x51			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_END_S *(mirror bits 0:3 of FP_V_SYNC_STRT_WID:FP_V_SYNC_WID)*	3:0	0x0	
V_INTR_CLR_M *(mirror of CRT11:V_INTR_CLR)*	4	0x0	0=VRetraceIntCleared 1=Not Cleared
V_INTR_EN_M *(mirror of CRT11:V_INTR_EN)*	5	0x0	0=VRetraceIntEna 1=Disable
SEL5_REFRESH_CYC_M *(mirror of CRT11:SEL5_REFRESH_CYC)*	6	0x0	0=3 DRAM Refresh/Horz Line 1=5 DRAM Refresh/Horz Line
C0T7_WR_ONLY_M *(mirror of CRT11:C0T7_WR_ONLY)*	7	0x0	0=EnaWrtToCRT00-07 1=C0T7B4WrtOnly

No description available for this register.

CRT12_S - RW - 8 bits - VGACRTIND:0x52			
Field Name	**Bits**	**Default**	**Description**
V_DISP_END_S *(mirror bits 0:7 of FP_CRTC_V_TOTAL_DISP:FP_CRTC_V_DISP)*	7:0	0x0	

No description available for this register.

CRT13_S - RW - 8 bits - VGACRTIND:0x53			
Field Name	**Bits**	**Default**	**Description**
DISP_PITCH_M *(mirror bits 0:7 of CRTC_PITCH:CRTC_PITCH)*	7:0	0x0	

No description available for this register.

CRT14_S - RW - 8 bits - VGACRTIND:0x54			
Field Name	**Bits**	**Default**	**Description**
UNDRLN_LOC_S (DISPLAY)	4:0	0x0	Shadow copy UNDRLN_N_LOC
ADDR_CNT_BY4_M *(mirror of CRT14:ADDR_CNT_BY4)* (DISPLAY)	5	0x0	Mirror of ADDR_CNT_BY4
DOUBLE_WORD_M *(mirror of CRT14:DOUBLE_WORD)* (HDP) (DISPLAY)	6	0x0	Mirror of DOUBLE_WORD

Shadow of Underline Location Register

CRT15_S - RW - 8 bits - VGACRTIND:0x55			
Field Name	**Bits**	**Default**	**Description**
V_BLANK_START_S	7:0	0x0	

No description available for this register.

CRT16_S - RW - 8 bits - VGACRTIND:0x56			
Field Name	**Bits**	**Default**	**Description**
V_BLANK_END_S	7:0	0x0	

No description available for this register.

CRT17_S - RW - 8 bits - VGACRTIND:0x57			
Field Name	**Bits**	**Default**	**Description**
RA0_AS_A13B_M *(mirror of CRT17:RA0_AS_A13B)* (DISPLAY)	0	0x0	Mirror of RA0_AS_A13B

RA1_AS_A14B_M *(mirror of CRT17:RA1_AS_A14B)* (DISPLAY)	1	0x0	Mirror of RA1_AS_A14B
VCOUNT_BY2_S (DISPLAY)	2	0x0	Shadow copy of VCOUNT_BY2
ADDR_CNT_BY2_M *(mirror of CRT17:ADDR_CNT_BY2)* (DISPLAY)	3	0x0	Mirror of ADDR_CNT_BY2
WRAP_A15TOA0_M *(mirror of CRT17:WRAP_A15TOA0)* (DISPLAY)	5	0x0	Mirror of WRAP_A15TOA0
BYTE_MODE_M *(mirror of CRT17:BYTE_MODE)* (HDP) (DISPLAY)	6	0x0	Mirror of BYTE_MODE
CRTC_SYNC_EN_M *(mirror of CRT17:CRTC_SYNC_EN)* (DISPLAY)	7	0x0	Mirror of CRTC_SYNC_EN

Shadow of CRT Mode Register

CRT18_S - RW - 8 bits - VGACRTIND:0x58			
Field Name	**Bits**	**Default**	**Description**
LINE_CMP_M *(mirror bits 0:7 of CRTC_VLINE_CRNT_VLINE:CRTC_VLINE)*	7:0	0x0	

No description available for this register.

2.20 vgaGrph Registers

GRPH8_IDX - RW - 8 bits - VGA_IO:0x3CE			
Field Name	**Bits**	**Default**	**Description**
GRPH_IDX (HDP) (DISPLAY)	3:0	0x0	This index is used to address one of the internal registers of the graphics controller (GRAC) at I/O port 0x3CRF.

GRPH Index Register

GRPH8_DATA - RW - 8 bits - VGA_IO:0x3CF			
Field Name	**Bits**	**Default**	**Description**
GRPH_DATA (HDP) (DISPLAY)	7:0	0x0	GRPH data indirect access

GRPH Data Register

GRA05 - RW - 8 bits - VGAGRPHIND:0x5			
Field Name	**Bits**	**Default**	**Description**
GRPH_WRITE_MODE (HDP)	1:0	0x0	Write Mode: 0=Write mode 0 1=Write mode 1 2=Write mode 2 3=Write mode 3
GRPH_READ1 (HDP)	3	0x0	Read Mode: 0=Read mode 0, byte oriented 1=Read mode 1, pixel oriented
CGA_ODDEVEN (HDP)	4	0x0	Odd/Even Addressing Enable. Used to enable CGA emulation, this bit enables off/even addressing mode when it is logical one. Normally, this bit and memory mode bit SEQ04[2] are set to agree with each other in enabling odd/even mode emulation. 0=Disable Odd/Even Addressing 1=Enable Odd/Even Addressing

GRPH_OES (HDP) (DISPLAY)	5	0x0	Shift Register Mode: This bit controls how data from memory is loaded into the shift registers M0D0:M0D7, M1D0:M1D7; M2D0:M2D7, and M3D0:M3D7 are representations of this data. 0=Linear shift mode 1=Tiled shift mode
GRPH_PACK (HDP) (DISPLAY)	6	0x0	256 Colour Mode. This bit also contros how data from memory is loaded into the shift registers. 0=Use shift register mode as per GRPH_OES 1=256 color mode, read as packed pixels, ignore GRPH_OES

Graphics Mode Register

2.21 vgaAttr Registers

ATTRX - RW - 8 bits - VGA_IO:0x3C0			
Field Name	**Bits**	**Default**	**Description**
ATTR_IDX	4:0	0x0	ATTR Index. This index points to one of the internal registers of the attribute controller (ATTR) at addresses 0x3C1/ 0x3C0, for the next ATTR read/write operation. Since both the index and data registers are at the same I/O, a pointer to the registers is necessary. This pointer cna be initialized to point to the index register by a read of GENS1.
ATTR_PAL_RW_ENB	5	0x0	Palette Address Source. After loading the colour palette, this bit should be set to logical 1. 0=Processor to load 1=Memory data to access

Attribute Index Register

ATTRDW - W - 8 bits - VGA_IO:0x3C0			
Field Name	**Bits**	**Default**	**Description**
ATTR_DATA	7:0	0x0	Attribute Data Write

Attribute Data Write Register

ATTRDR - R - 8 bits - VGA_IO:0x3C1			
Field Name	**Bits**	**Default**	**Description**
ATTR_DATA *(mirror of ATTRDW:ATTR_DATA)*	7:0	0x0	Attribute Data Read

Attribute Index Register

ATTR00 - RW - 8 bits - VGAATTRIND:0x0			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 0

ATTR01 - RW - 8 bits - VGAATTRIND:0x1			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 1

ATTR02 - RW - 8 bits - VGAATTRIND:0x2			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 2

ATTR03 - RW - 8 bits - VGAATTRIND:0x3			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 3

ATTR04 - RW - 8 bits - VGAATTRIND:0x4			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 4

ATTR05 - RW - 8 bits - VGAATTRIND:0x5			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 5

ATTR06 - RW - 8 bits - VGAATTRIND:0x6			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 6

ATTR07 - RW - 8 bits - VGAATTRIND:0x7			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 7

ATTR08 - RW - 8 bits - VGAATTRIND:0x8			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 8

ATTR09 - RW - 8 bits - VGAATTRIND:0x9			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register 9

ATTR0A - RW - 8 bits - VGAATTRIND:0xA			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register Ah (10)

ATTR0B - RW - 8 bits - VGAATTRIND:0xB			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register Bh (11)

ATTR0C - RW - 8 bits - VGAATTRIND:0xC			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register Ch (12)

ATTR0D - RW - 8 bits - VGAATTRIND:0xD			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register Dh (13)

ATTR0E - RW - 8 bits - VGAATTRIND:0xE			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register Eh (14)

ATTR0F - RW - 8 bits - VGAATTRIND:0xF			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL	5:0	0x0	Colour Bits 5:0 map the text attribute or graphics colour input value to a display colour on the screen. Colour is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.

Palette Register Fh (15)

ATTR10 - RW - 8 bits - VGAATTRIND:0x10			
Field Name	**Bits**	**Default**	**Description**
ATTR_GRPH_MODE	0	0x0	Graphics/Alphanumeric Mode. 0=Alphanumeric Mode 1=Graphic Mode
ATTR_MONO_EN	1	0x0	Monochrome/Colour Attributes Select: 0=Color Disp 1=MonoChrome Disp
ATTR_LGRPH_EN	2	0x0	Line Graphics Enable. Must be 0 for character fonts that do not use line graphics character codes for graphics. Zero will force the 9th dot to the background colour. One will allow the 8th bit of the line graphics characters to be stretched to the 9th dot. 0=Disable line graphics 8th dot stretch 1=Enable line graphics 8th dot stretch
ATTR_BLINK_EN	3	0x0	Blink Enable/Background Intensity: Selects whether bit 7 of the attribute controls intensity or blinking. 0=Intensity control 1=Blink control
ATTR_PANTOPONLY	5	0x0	PEL Panning Compatibility: 0=Panning both 1=Panning only the top half screen
ATTR_PCLKBY2	6	0x0	PEL Clock Select: 0=Shift register clocked every dot clock 1=For mode 13 (256 colour), 8 bits packed to form a pixel
ATTR_CSEL_EN	7	0x0	Alternate Colour Source: 0=Select ATTR00-0F bit 5:4 as P5 and P4 1=Select ATTR14 bit 1:0 as P5 and P4

Mode Control Register

ATTR11 - RW - 8 bits - VGAATTRIND:0x11			
Field Name	**Bits**	**Default**	**Description**
ATTR_OVSC	7:0	0x0	Overscan Colour

Overscan Colour Register

ATTR12 - RW - 8 bits - VGAATTRIND:0x12			
Field Name	**Bits**	**Default**	**Description**
ATTR_MAP_EN	3:0	0x0	Enable Colour Map bits. 0 = Disables data from respective map from being used for video output. 1 = Enables data from respective map for use in video output.
ATTR_VSMUX	5:4	0x0	Video Status Mux bits 1:0. These are control bits for the multiplexer on colour bits P0-P7. The bit selection is also indicated at GENS1[5:4]: 00 = P2, P0 01 = P5, P4 10 = P3, P1 11 = P7, P6

Colour Map Enable Register

ATTR13 - RW - 8 bits - VGAATTRIND:0x13			
Field Name	**Bits**	**Default**	**Description**
ATTR_PPAN	3:0	0x0	Shift Count Bits 3:0. The shift count value (0-8) indicates how many pixle positions to shift left. Shift in respective modes Count 0+,1+,2+, 13 All other Value 3+,7,7+ 0 1 0 0 1 2 - 1 2 3 1 2 3 4 - 3 4 5 2 4 5 6 - 5 6 7 3 6 7 8 - 7 8 0 - -

Horizontal PEL Panning Register

ATTR14 - RW - 8 bits - VGAATTRIND:0x14			
Field Name	**Bits**	**Default**	**Description**
ATTR_CSEL1	1:0	0x0	Colour bits P5 and P4, respectively. These are the colour output bits (instead of bits 5 and 4 of the internal palette registers ATTR00-0F) when alternate colour source, bit ATTR10[7] is logical 1.
ATTR_CSEL2	3:2	0x0	Colour bits P7 and P6, respectively. These two bits are the two high-order bits of the 8-bit colour, used for rapid colour set switching (addressing different parts of the DAC colour lookup table). The lower order bits are in registers ATTR00-0F.

Colour Select Register

2.22 CRTC Registers

CRTC_H_TOTAL_DISP - RW - 32 bits - [MMReg:0x200]			
Field Name	**Bits**	**Default**	**Description**
CRTC_H_TOTAL	9:0	0x0	Horizontal total (pixels * 8)-1. Sum of display width, overscan right, front porch, sync width, back porch and overscan left. This field is programmed with the -1 from the desired size. i.e. for 640 pixels set to 800/8 - 1 = 99
CRTC_H_DISP	24:16	0x0	Horizontal display end (pixels * 8)-1. Determines number of visible pixels, not including overscan. This field is programmed with the -1 from the desired size. i.e. for 640 pixels set to 640/8 - 1 = 79

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

CRTC_H_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x204]			
Field Name	**Bits**	**Default**	**Description**
CRTC_H_SYNC_STRT_PIX	2:0	0x0	CRT Horizontal sync start delay in pixels. Allows pixel accurate horizontal positioning by delaying sync position within character position set below.
CRTC_H_SYNC_STRT_CHAR	12:3	0x0	Horizontal sync start (pixels * 8). Sum of display width, overscan right and front porch. This field is programmed with the exact desired character on which to start HSYNC, not -1. i.e. 656/8 = 82
CRTC_H_SYNC_WID	21:16	0x0	Horizontal sync width (pixels * 8)
CRTC_H_SYNC_POL	23	0x0	CRT Horizontal sync polarity 0 = Active high 1 = Active Low 0=Active high 1=Active low
CRTC_H_SYNC_SKEW_TUNE	26:24	0x0	Per pixel HSYNC skew tuning value used when manual tuning selected. Value is per display pixel. For VGA display adjustment range varies with VGA mode. For extended display the range is -1 (0x0) to +6 (0x7) pixels.

CRTC_H_SYNC_SKEW_TUNE_MODE	28	0x0	Used mostly in VGA mode to have HW automatically adjust HSYNC position pixel accurately based on current VGA mode. 0=Auto tune the DAC HSYNC pixel skew. 1=Manually tune the DAC HSYNC pixel skew using CRTC_H_SYNC_SKEW_TUNE.

Horizontal Sync Control.
The CRTC_H_SYNC_STRT_PIX and CRTC_H_SYNC_POL always apply to the CRT Horizontal Sync regardless of other control bits.(DFP_SYNC_SEL/CRT_SYNC_SEL/CRT_SYNC_ALT_SEL/horz. blanking mode/horz. autocentering)

CRTC_V_TOTAL_DISP - RW - 32 bits - [MMReg:0x208]			
Field Name	**Bits**	**Default**	**Description**
CRTC_V_TOTAL	11:0	0x0	Vertical total (lines-1). Sum of display height, overscan bottom, front porch, sync width, back porch and overscan top. This field is set to one less than the desired number of total lines.
CRTC_V_DISP	27:16	0x0	Vertical display end (lines-1). Determines number of visible lines, not including overscan. This field is set to one less than the desired number of visible lines.

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

CRTC_V_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x20C]			
Field Name	**Bits**	**Default**	**Description**
CRTC_V_SYNC_STRT	11:0	0x0	Vertical sync start. Sum of display height, overscan bottom and front porch. This field is set the exact line on which VSYNC should start, i.e. not -1 like for CRTC_V_TOTAL or CRTC_V_DISP.
CRTC_V_SYNC_WID	20:16	0x0	Vertical sync width
CRTC_V_SYNC_POL	23	0x0	Vertical sync polarity 0 = Active high 1 = Active low 0=Active high 1=Active low

Vertical Sync Control

CRTC_VLINE_CRNT_VLINE - RW - 32 bits - [MMReg:0x210]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VLINE	11:0	0x0	Vertical line at which vertical line interrupt is triggered.
CRTC_CRNT_VLINE (R)	27:16	0x0	Current vertical line.

Display Current Vertical Line

CRTC_CRNT_FRAME - R - 32 bits - [MMReg:0x214]			
Field Name	**Bits**	**Default**	**Description**
CRTC_CRNT_FRAME	20:0	0x0	Readback of current value of display frame counter. Used by display time sensitive applications such as video playback.

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

CRTC_GUI_TRIG_VLINE - RW - 32 bits - [MMReg:0x218]			
Field Name	**Bits**	**Default**	**Description**
CRTC_GUI_TRIG_VLINE_START	11:0	0x0	The START (upper in display, lower in memory) for the GUI_TRIG_VLINE compare. First line of display is line 0.
CRTC_GUI_TRIG_VLINE_INV	15	0x0	Controls whether VLINE region signal is active high or active low. Can be used to stall command stream parser until inside region, or not inside region. 0=Active when raster between START and END. 1=Active when raster outside START and END.
CRTC_GUI_TRIG_VLINE_END	27:16	0x0	The END (lower in display, higher in memory) for the GUI_TRIG_VLINE compare. First line of display is line 0.
CRTC_GUI_TRIG_VLINE_STALL	30	0x0	If waiting on rising or falling edge of VLINE using WAIT_UNTIL register, then always set the stall to 0. If using WAIT_UNTIL VLINE=1, then always set this stall bit to 1. This bit controls the timing of the signal from the display to the command stream parser during writes to this register. 0=Normal operation. 1=Force low during write of this register.

CRTC_GUI_TRIG_VLINE (R)	31	0x0	This signal is active active when the raster is between the START and END. START <= raster <= END. The polarity is controlled as above. This signal goes to the command stream parser as a condition in the WAIT_UNTIL register. 0=Current line not between VLINE start and end. 1=Current line is between VLINE start and end, inclusive.

Trigger to GUI engine activated in certain vertical region of the display, when the raster is between START and END. Normally used to delay rendering operations until the raster has passed a specific point.

CRTC_DEBUG - RW - 32 bits - [MMReg:0x21C]			
Field Name	**Bits**	**Default**	**Description**
CRTC_GUI_TRIG_BYPASS_EN	0	0x0	Enables manual testing of display GUI trigger signals to the WAIT_UNTIL register and the render backend. 0=Don't bypass gui triggers generated by dispeng. 1=Bypass gui triggers generated by dispeng.
GUI_TRIG_VLINE_BYPASS	1	0x0	When GUI triggers are bypassed, this sets the value of the VLINE status signal to the WAIT_UNTIL register.
GUI_TRIG_OFFSET_BYPASS	2	0x0	When GUI triggers are bypassed, this sets the value of the CRTC_OFFSET status signal to the WAIT_UNTIL register.
GUI_TRIG_PITCH_ADD_BYPASS	3	0x0	When GUI triggers are bypassed, this sets the value of the CRTC_PITCH_ADD status signal to the rneder backend (this is for rendering stall behind the raster, which Rage 6 render backend may not support).

Controls for HW testing and debug of display systems. Should not be needed by software.

CRTC_OFFSET_RIGHT - RW - 32 bits - [MMReg:0x220]			
Field Name	**Bits**	**Default**	**Description**
CRTC_OFFSET_RIGHT	26:0	0x0	Graphics surface origin offset in memory for right eye. Alignment must be as for CRTC_OFFSET. Used only when CRTC_STEREO_OFFSET_EN = 1. This is added to DISP_BASE_ADDR to get the full 32 bit address of the graphics surface. NOTE: Bits 0:2 of this field are hardwired to ZERO.
CRTC_GUI_TRIG_OFFSET *(mirror of CRTC_OFFSET:CRTC_GUI_TRIG_OFFSET)* (R)	30	0x0	Indicates if visible buffer is last written, or still the previous one. See CRTC_OFFSET register. 0=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written is being displayed 1=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written not yet displayed

CRTC_OFFSET_LOCK *(mirror of CRTC_OFFSET:CRTC_OFFSET_LOCK)*	31	0x0	Prevents hardware from internally updating certain fields until cleared. See CRTC_OFFSET register. 0=Unlock these regs 1=Lock'em

Right eye visible surface origin. Only used in stereo

CRTC_OFFSET - RW - 32 bits - [MMReg:0x224]			
Field Name	**Bits**	**Default**	**Description**
CRTC_OFFSET	26:0	0x0	Graphics surface origin offset in memory. Must be 8 byte aligned for 4bpp, 8bpp and 24bpp modes. Must be 16 byte aligned for all 16bpp modes. Must be 32 byte aligned for 32bpp mode. When CRTC_STEREO_OFFSET_EN = 1, this is the left eye image. This is added to DISP_BASE_ADDR to get the full 32 bit address of the graphics surface. NOTE: Bits 0:2 of this field are hardwired to ZERO.
CRTC_GUI_TRIG_OFFSET (R)	30	0x0	Indicates if visible buffer is last written, or still the previous one. This bit is read only. Goes high when an offset has been written but the corresponding buffer does not appear on screen yet. It goes low again when display starts for that address. 0=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written is being displayed 1=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written not yet displayed
CRTC_OFFSET_LOCK	31	0x0	Prevents hardware from internally updating the following fields until cleared: CRTC_OFFSET, CRTC_OFFSET_RIGHT, CRTC_TILE_LINE, CRTC_TILE_LINE_RIGHT. It permits atomic update of CRTC_OFFSET and CRTC_TILE_LINE. Normal operation is with the lock in zero. 0=Unlock these regs 1=Lock'em

Main graphics visible surface origin. Left eye image for stereo

CRTC_OFFSET_CNTL - RW - 32 bits - [MMReg:0x228]			
Field Name	**Bits**	**Default**	**Description**
CRTC_TILE_LINE	3:0	0x0	When CRTC_TILE_EN = 1, this field holds the 4 LSB of the line of the surface where CRTC_OFFSET starts (the 'y' for line 0 of the display, or 'start line'). The display address generator needs to know this to determine the proper pitch to add at the end of each display line. This is normally 0, unless the display is in a virtual desktop mode with tiling enabled. For example, if the surface offset is zero and the display starts in line 3, CRTC_TILE_LINE=0x3 CRTC_OFFSET = 0xC0 Note that tiles are 8 lines high, but this register must contain ((start line) MOD 16) in order to do the checkerboarding correctly. Do not worry about what checkerboarding is, you shouldn't need to know.
CRTC_TILE_LINE_RIGHT	7:4	0x0	When CRTC_TILE_EN_RIGHT = 1, this field holds the 4 LSB of the line of the surface where CRTC_OFFSET_RIGHT starts. Same as for the left eye with CRTC_TILE_LINE and CRTC_OFFSET. Right fields only used when CRTC_STEREO_OFFSET_EN = 1.
CRTC_TILE_EN_RIGHT	14	0x0	Graphics display tiling enable for right eye image. Used only when CRTC_STEREO_OFFSET_EN = 1. Tiling is not supported and should not be enabled in double scan or interlaced display modes. i.e. CRTC_DBL_SCAN_EN=1 or CRTC_INTERLACE_EN=1 0=Display Surface uses linear addressing 1=Display surface uses tiled addressing
CRTC_TILE_EN	15	0x0	Graphics display tiling enable. When CRTC_STEREO_OFFSET_EN = 1 controls tiling of left eye image only. Tiling is not supported and should not be enabled in double scan or interlaced display modes. i.e. CRTC_DBL_SCAN_EN=1 or CRTC_INTERLACE_EN=1 0=Display Surface uses linear addressing 1=Display surface uses tiled addressing
CRTC_OFFSET_FLIP_CNTL	16	0x0	Selects position within the frame at which new CRTC_OFFSET will be used. Should be normally zero. If set to one, a new offest will be taken at the end of the line instead of the end of the frame. 0=Use new CRTC_OFFSET on vertical blank 1=Use new CRTC_OFFSET on any horizontal blank. Note, this can cause the display to tear.

CRTC_STEREO_OFFSET_EN	17	0x0	Enables the use of the stereo display right eye offset and pitch fields. 0=No stereoscopic. RIGHT registers not used 1=Stereoscopic display enabled. Alternate display between normal (left) and right display surfaces
CRTC_STEREO_SYNC_EN	19:18	0x0	Controls function of STEREOSYNC output signal. 0=STEREOSYNC always low 1=STEREOSYNC alternates. Low=right, high=left 2=STEREOSYNC always high 3=STEREOSYNC alternates. Low=left, high=right
CRTC_STEREO_SYNC_OUT_EN	20	0x0	The output enable for the STEREOSYNC output signal. 0=STEREOSYNC output tri-stated 1=STEREOSYNC output enabled
CRTC_STEREO_SYNC (R)	21	0x0	Readback of STEREOSYNC signal current value.
CRTC_GUI_TRIG_OFFSET_LEFT_EN	28	0x1	Controls whether or not CRTC_OFFSET affects the CRTC_GUI_TRIG_OFFSET to the WAIT_UNTIL register. If enabled, then wait condition is set on writing CRTC_OFFSET, and cleared only if display starts displaying from CRTC_OFFSET (not if display uses CRTC_OFFSET_RIGHT for next frame). 0=Writing CRTC_OFFSET (main/left) does not set CRTC_GUI_TRIG_OFFSET 1=Writing CRTC_OFFSET sets CRTC_GUI_TRIG_OFFSET
CRTC_GUI_TRIG_OFFSET_RIGHT_EN	29	0x0	Controls whether or not CRTC_OFFSET_RIGHT affects the CRTC_GUI_TRIG_OFFSET to the WAIT_UNTIL register. If enabled, then wait condition is set on writing CRTC_OFFSET_RIGHT, and cleared only if display starts displaying from CRTC_OFFSET_RIGHT (not if display uses CRTC_OFFSET for next frame). 0=Writing CRTC_OFFSET_RIGHT does not set CRTC_GUI_TRIG_OFFSET 1=Writing CRTC_OFFSET_RIGHT sets CRTC_GUI_TRIG_OFFSET
CRTC_GUI_TRIG_OFFSET *(mirror of CRTC_OFFSET:CRTC_GUI_TRIG_OFFSET)* (R)	30	0x0	Indicates if visible buffer is last written, or still the previous one. See CRTC_OFFSET register. 0=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written is being displayed 1=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written not yet displayed
CRTC_OFFSET_LOCK *(mirror of CRTC_OFFSET:CRTC_OFFSET_LOCK)*	31	0x0	Prevents hardware from internally updating certain fields until cleared. See CRTC_OFFSET register. 0=Unlock these regs 1=Lock'em

Graphics Display Address Generator Control

CRTC_PITCH - RW - 32 bits - [MMReg:0x22C]			
Field Name	**Bits**	**Default**	**Description**
CRTC_PITCH	10:0	0x0	Display line pitch in (pixels * 8). Note that for 24bpp the display uses pixels * 8 for the pitch, but the rendering engine uses bytes * 8 for the pitch. For tiled display this is the same pitch as used for the surface in the rendering engine (24bpp not supported for tiled). In tiled the pitch must be a multiple of 256 bytes (the tile width). So for 32bpp tiled, the CRTC_PITCH must be a multiple of 64 pixels. Or 128 pixels for 16bpp, or 256 pixels for 8bpp. For stereo display mode, this is the left image pitch.
CRTC_PITCH_RIGHT	26:16	0x0	Programming details same as for CRTC_PITCH. This field only used in stereo display mode for right image pitch.

Graphics Display Address Pitch

CRT_CRTC_H_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x258]			
Field Name	**Bits**	**Default**	**Description**
CRT_CRTC_H_SYNC_STRT_CHAR	12:3	0x0	Horizontal sync start position in characters(pixels * 8). Sum of display width, overscan right and front porch. This field is programmed with the exact desired character on which to start HSYNC, not -1. i.e. 656/8 = 82 If horizontal autocentering or fixed blank is enabled(see CRTC_MORE_CNTL) this field contains the offset (from the end of the active display/start of horizontal blank) within the blank time of the start of the horizontal sync signal for the CRT (in characters)
CRT_CRTC_H_SYNC_WID	21:16	0x0	Horizontal sync width in characters(pixels * 8) If horz. autocentering or fixed blank is enabled, this field contains the width (in characters) of the horizontal sync signal for the CRT.

Alternate Horizontal CRT Sync.
CRT HSync generated using this value when FP_GEN_CNTL.CRT_ALT_SYNC_SEL = 1

CRT_CRTC_V_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x25C]			
Field Name	**Bits**	**Default**	**Description**
CRT_CRTC_V_SYNC_STRT	11:0	0x0	Vertical sync start. Sum of display height, overscan bottom and front porch. If vertical autocentering or fixed blank is enabled(see CRTC_MORE_CNTL) this field contains the offset (from the bottom of the active display/start of vertical blank) within the vertical blank time of the start of the vertical sync signal for the CRT (in lines)
CRT_CRTC_V_SYNC_WID	20:16	0x0	Vertical sync width (in lines). If vertical autocentering or fixed blank is enabled this field contains the length (in lines) of the vertical sync signal for the CRT

Alternate Vertical CRT Sync.
CRT VSync generated using this value when FP_GEN_CNTL.CRT_ALT_SYNC_SEL = 1

CRTC_MORE_CNTL - RW - 32 bits - [MMReg:0x27C]			
Field Name	**Bits**	**Default**	**Description**
CRTC_HORZ_BLANK_MODE_SEL	0	0x0	Selects horizontal retrace mode. Either traditional (0) or fixed blank (1). 0=CRTC H blank timings are relative to start of active. 1=CRTC H blank timings are relative to end of active.
CRTC_VERT_BLANK_MODE_SEL	1	0x0	Selects vertical retrace mode. Either traditional (0) or fixed blank (1). 0=CRTC V blank timings are relative to start of active. 1=CRTC V blank timings are relative to end of active.
CRTC_AUTO_HORZ_CENTER_EN	2	0x0	Enables horizontal auto-centering circuit. See FP_CRTC_H_TOTAL_DISP, FP_CRTC_V_TOTAL_DISP, CRT_CRTC_H_SYNC_STRT_WID, CRT_CRTC_V_SYNC_STRT_WID, FP_H_SYNC_STRT_WID, and FP_V_SYNC_STRT_WID. 0=H centering is manual. 1=H centering is automatic based on H_DISP and FP_HORZ_PANEL_SIZE.
CRTC_AUTO_VERT_CENTER_EN	3	0x0	Enables vertical auto-centering circuit. 0=V centering is manual. 1=V centering is automatic based on V_DISP and FP_VERT_PANEL_SIZE.

CRTC_H_CUTOFF_ACTIVE_EN	4	0x0	Enables horizontal active size forcing circuit. Uses FP_HORZ_STRETCH.FP_HORZ_PANEL_SIZE 0=No corrections to CRTC and/or RMX horizontal timings. 1=Horizontal active forced to FP_HORZ_PANEL_SIZE by stretching or cutting off CRTC timings as needed.
CRTC_V_CUTOFF_ACTIVE_EN	5	0x0	Enables vertical active size forcing circuit. Uses FP_VERT_STRETCH.FP_VERT_PANEL_SIZE 0=No corrections to CRTC and/or RMX vertical timings. 1=Vertical active forced to FP_VERT_PANEL_SIZE by stretching or cutting off CRTC timings as needed.

More CRTC controls
For 'fixed blank' and 'autocentering' modes, refer to registers FP_CRTC_H_TOTAL_DISP, FP_CRTC_V_TOTAL_DISP, FP_H_SYNC_STRT_WID, FP_V_SYNC_STRT_WID, CRT_CRTC_H_SYNC_STRT_WID, and CRT_CRTC_V_SYNC_STRT_WID. For 'autocentering' please also refer to FP_HORZ_VERT_ACTIVE. When autocentering is enabled, fixed blanking is also enabled.

CRTC_GEN_CNTL - RW - 32 bits - [IOReg,MMReg:0x50]			
Field Name	**Bits**	**Default**	**Description**
CRTC_DBL_SCAN_EN (DISPLAY)	0	0x0	Double scan enable. Double scan only affects the calculation of display addresses by adding the CRTC_PITCH every second line (also applies to the hardware cursor pitch). Enabling double scan does not change the CRTC vertical programming or VSYNC timing. The overscan top & bottom are not affected and remain the number of lines programmed (i.e. not double). The hardware cursor programming remains in terms of physical lines (not logical lines). The cursor vertical position must begin on an even line number when in double scan. The cursor itself is limited to 64 physical lines in height, which means only 32 logical lines. This is because the cursor pitch is only added at the end of odd scan lines, but the CRTC vertical logic stops the cursor after 64 physical lines. 0=disable 1=enable
CRTC_INTERLACE_EN (DISPLAY)	1	0x0	Interlace display mode enable. 0=Non-Interlace 1=Interlace
CRTC_C_SYNC_EN (DISPLAY)	4	0x0	Enables composite sync on horizontal sync output. When this is set, the VSYNC pin should be disabled by setting CRTC_EXT_CNTL.CRTC_VSYNC_TRISTATE=1. 0=Disable 1=Enable

CRTC_PIX_WIDTH (DISPLAY)	11:8	0x0	Display pixel width (actually depth). For 4bpp mode DAC_CNTL.DAC_4BPP_PIX_ORDER selects the order of nibbles within bytes. When R, G, or B components are only 4, 5 or 6 bits, DAC_CNTL.DAC_EXPAND_MODE selects how these components are expanded to 8 bits each for keying and palette lookup. When alpha values are 1 or 4 bits, they are expanded to 8 bits by dynamic expansion of the high order bits to the missing lower order bits. 0=Disable pixel clock for primary CRTC 1=4bpp Indexed 2=8bpp Indexed 3=15bpp aRGB 1555 4=16bpp RGB 565 5=24bpp RGB 888 6=32bpp aRGB 8888 7=16bpp aRGB 4444 8=16bpp aIndex 88
CRTC_ICON_EN (DISPLAY)	15	0x0	0=Disable Hardware Icon 1=Enable Hardware Icon
CRTC_CUR_EN (DISPLAY)	16	0x0	Hardware cursor enable. This field is double buffered and locked with the CUR_LOCK register field. 0=Disable 1=Enable

CRTC_VSTAT_MODE (DISPLAY)	18:17	0x0	Selects the location of the display updating of CRTC_OFFSET and CUR_OFFSET and related fields during the vertical retrace. Also determines where the VGA_VSTATUS update will occur within the vertical retrace for VGA modes. For non-VGA modes only the upper bit of this field is used. When 00 or 01 in non-VGA the updating of the CRTC_OFFSET and the CUR_OFFSET is delayed as long as possible within the vertical retrace until the start of the last line of the retrace. When 10 or 11 in non-VGA the CRTC_OFFSET and CUR_OFFSET are updated at the leading edge of VSYNC, which is normally relatively early in the vertical retrace. No matter how this is set, the update of status bits like WAIT_UNTIL_PFLIP will reflect the actual location of the update. Any writes to CRTC_OFFSET or CUR_OFFSET after the selected point has passed in the vertical retrace will have no affect until the display frame after the next vertical retrace. For VGA modes this field affects the behaviour of page flipping in some applications/games that poll the VGA_VSTATUS flag. This field should be tuned by the BIOS for compatibility with the most games. 0=VGA_VSTATUS until vcount= vtotal, DISP_ADDR loads when vcount=vt otal 1=VGA_VSTATUS until vblank end, DISP_ADDR loads when vcount=vtotal 2=VGA_VSTATUS until vcount= vtotal,DISP_ADDR loads in vsync start 3=VGA_VSTATUS until vblank end,DISP_ADDR loads in vsync start
CRTC_CUR_MODE (DISPLAY)	22:20	0x0	Hardware cursor mode. For 2bpp mode, each line of cursor data is stored in memory as 64 bits of AND data followed by 64 bits or XOR data. For color AND/XOR mode, each pixel is stored sequentially in memory as 32bits each in aRGB8888 format with bit 31 of each DWord being the AND bit. For the color alpha modes the format is also 32bpp aRGB8888 with all 8 bits of the alpha being used. All HW cursor lines must be 64 pixels wide and all lines must be stored sequentially in memory. 0=Mono + 1 bit AND (2bpp), 64x64 1=Color 24bpp + 1 bit AND, 32hx64v 2=Color 24bpp + 8 bit aplha, premultiplied alpha, 32hx64v 3=Color 24bpp + 8 bit aplha, unmultiplied alpha, 32hx64v
CRTC_EXT_DISP_EN (BIF) (DISPLAY)	24	0x0	Extended display mode enable. No affect if strapped VGA_DISABLE=1. 0=VGA 1=Extended

CRTC_EN (DISPLAY)	25	0x0	Enables CRT controller. When reset, the CRTC horizontal counter is set to zero, and the vertical counter is set to the current value of CRTC_V_TOTAL_DISP.CRTC_V_DISP. 0=Reset 1=Enable
CRTC_DISP_REQ_EN_B (DISPLAY)	26	0x1	Enables display graphics requests to the memory controller. Affects only graphics and VGA text requests. Does not affect HW cursor, overlay or subpic. When setting this bit, CRTC_EXT_CNTL.CRTC_DISPLAY_DIS is also normally set to blank the screen. Active low. 0=Enable 1=Disable

CRTC general controls

CRTC_EXT_CNTL - RW - 32 bits - [IOReg,MMReg:0x54]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VGA_XOVERSCAN (DISPLAY)	0	0x0	Set low for VGA compatible borders. When set high extended overscan registers control border in VGA modes. See also auto-centering in CRTC_MORE_CNTL and FP_GEN_CNTL.CRTC_VGA_XOVERSCAN_COLOR. 0=Disable extended overscan in VGA 1=Enable extended overscan in VGA
VGA_BLINK_RATE (DISPLAY)	2:1	0x0	Controls number of frames per blink for VGA modes. 0=Default VGA blink rate (16 frames) 1=1/2 default VGA blink rate (32 frames) 2=1/3 default VGA blink rate (48 frames) 3=1/4 default VGA blink rate (64 frames)
VGA_ATI_LINEAR (HDP) (DISPLAY)	3	0x0	Enable linear addressing through VGA memory aperture. 0=Disable 1=Enable
VGA_128KAP_PAGING (HDP) (DISPLAY)	4	0x0	Enable extended aperture paging in 128K VGA aperture mode. 0=Normal 1=Enable
VGA_TEXT_132 (HDP) (DISPLAY)	5	0x0	Extended text mode select (linear address 132 column text mode). Set low for VGA compatible 40 or 80 column text modes. 0=inActive 1=Active

VGA_XCRT_CNT_EN (DISPLAY)	6	0x0	Extended CRTC display address counter enable. Active High 0=Disable 1=Enable Ext CRTC Counter
CRTC_HSYNC_DIS (DISPLAY)	8	0x0	Disables horizontal sync output. Could be used for DPMS signaling, but DISP_PWR_MAN.DISP_PWR_MAN_DPMS is recommended instead. 0=Enable 1=Disable
CRTC_VSYNC_DIS (DISPLAY)	9	0x0	Disables vertical sync output. Could be used for DPMS signaling, but DISP_PWR_MAN.DISP_PWR_MAN_DPMS is recommended instead. Note this must remain enabled while using composite SYNC on HSYNC (CRTC_C_SYNC_EN=1). 0=Enable 1=Disable
CRTC_DISPLAY_DIS (DISPLAY)	10	0x0	Disables the display, forcing the blanking signal to be active. When blanking the screen with this bit, the overlay, sub-picture, graphics and cursor should also be disabled to save power. See CRTC_DISP_REQ_EN_B, CRTC_CUR_EN, OV0_OVERLAY_EN and SUBPIC_ON. 0=Enable 1=Blanked
CRTC_SYNC_TRISTATE (DISPLAY)	11	0x0	Tristates HSYNC and VSYNC outputs. For individual tristate control, see the next two fields. 0=Normal 1=Tristate HSYNC and VSYNC outputs
CRTC_HSYNC_TRISTATE (DISPLAY)	12	0x0	Tristates HSYNC output. 0=Normal HSYNC 1=Tristate HSYNC output
CRTC_VSYNC_TRISTATE (DISPLAY)	13	0x0	Tristates VSYNC output. This is recommended for use with composite sync mode when only the HSYNC output pin should be enabled. 0=Normal VSYNC 1=Tristate VSYNC output
CRT_ON (DISPLAY)	15	0x0	0=CRT OFF 1=CRT ON
VGA_CUR_B_TEST (DISPLAY)	17	0x0	Test cursor blinking. Only used for diagnostic testing. 0=Disable VGA cursor test 1=Test VGA cursor blinking
VGA_PACK_DIS (HDP)	18	0x0	Controls host write pipe for packed VGA modes (e.g. mode 13). Should only be set high if HW problem with fast writes. 0=Fast VGA write in packed modes 1=Normal VGA write in packed modes

VGA_MEM_PS_EN (HDP)	19	0x0	VGA page select enable: 0=Don't use MEM_VGA_WP_SEL and MEM_VGA_RP_SEL registers 1=Use MEM_VGA_WP_SEL and MEM_VGA_RP_SEL registers
VCRTC_IDX_MASTER (HDP) (DISPLAY)	30:24	0x0	VGA CRTC master index. Only bits 5:0 of the VGA CRTC index can be written (or read) in VGA I/O space at 0x3B4 or 0x3D4. Bit 6 controls whether the master or shadow set of VGA CRTC registers is seen in VGA I/O space. The shadow set is for use when supporting panel operation in VGA modes. The BIOS will leave either the master or shadow set active as needed after a mode switch call.

More CRTC general controls

CRTC2_H_TOTAL_DISP - RW - 32 bits - [MMReg:0x300]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_H_TOTAL	9:0	0x0	Horizontal total (pixels * 8)-1. Sum of display width, overscan right, front porch, sync width, back porch and overscan left. This field is programmed with the -1 from the desired size. i.e. for 640 pixels set to 800/8 - 1 = 99
CRTC2_H_DISP	24:16	0x0	Horizontal display end (pixels * 8)-1. Determines number of visible pixels, not including overscan. This field is programmed with the -1 from the desired size. i.e. for 640 pixels set to 640/8 - 1 = 79

Secondary Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

CRTC2_H_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x304]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_H_SYNC_STRT_PIX	2:0	0x0	CRT Horizontal sync start delay in pixels. Allows pixel accurate horizontal positioning by delaying sync position within character position set below.
CRTC2_H_SYNC_STRT_CHAR	12:3	0x0	Horizontal sync start (pixels * 8). Sum of display width, overscan right and front porch.
CRTC2_H_SYNC_WID	21:16	0x0	Horizontal sync width (pixels * 8)

CRTC2_H_SYNC_POL	23	0x0	CRT Horizontal sync polarity 0 = Active high 1 = Active Low 0=Active high 1=Active low

Horizontal Sync Control.
The CRTC2_H_SYNC_STRT_PIX and CRTC2_H_SYNC_POL always apply to the CRT Horizontal Sync regardless of other control bits.(DFP_SYNC_SEL/CRT_SYNC_SEL/CRT_SYNC_ALT_SEL/horz. blanking mode/horz. autocentering)

CRTC2_V_TOTAL_DISP - RW - 32 bits - [MMReg:0x308]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_V_TOTAL	11:0	0x0	Vertical total. Sum of display height, overscan bottom, front porch, sync width, back porch and overscan top.
CRTC2_V_DISP	27:16	0x0	Vertical display end. Determines number of visible lines, not including overscan.

Vertical Total Control

CRTC2_V_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x30C]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_V_SYNC_STRT	11:0	0x0	Vertical sync start. Sum of display height, overscan bottom and front porch.
CRTC2_V_SYNC_WID	20:16	0x0	Vertical sync width
CRTC2_V_SYNC_POL	23	0x0	Vertical sync polarity 0 = Active high 1 = Active low 0=Active high 1=Active low

Vertical Sync Control

CRTC2_VLINE_CRNT_VLINE - RW - 32 bits - [MMReg:0x310]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_VLINE	11:0	0x0	Vertical line at which vertical line interrupt is triggered.
CRTC2_CRNT_VLINE (R)	27:16	0x0	Current vertical line.

Display Current Vertical Line

CRTC2_CRNT_FRAME - R - 32 bits - [MMReg:0x314]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_CRNT_FRAME	20:0	0x0	Readback of current value of display frame counter. Used by display time sensitive applications such as video playback.

Overscan color. Always 24 bit, independent of pixel depth.

CRTC2_GUI_TRIG_VLINE - RW - 32 bits - [MMReg:0x318]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_GUI_TRIG_VLINE_START	11:0	0x0	The START (upper in display, lower in memory) for the GUI_TRIG_VLINE compare. First line of display is line 0.
CRTC2_GUI_TRIG_VLINE_INV	15	0x0	Controls whether VLINE region signal is active high or active low. Can be used to stall command stream parser until inside region, or not inside region. 0=Active when raster between START and END. 1=Active when raster outside START and END.
CRTC2_GUI_TRIG_VLINE_END	27:16	0x0	The END (lower in display, higher in memory) for the GUI_TRIG_VLINE compare. First line of display is line 0.
CRTC2_GUI_TRIG_VLINE_STALL	30	0x0	If waiting on rising or falling edge of VLINE using WAIT_UNTIL register, then always set the stall to 0. If using WAIT_UNTIL VLINE=1, then always set this stall bit to 1. This bit controls the timing of the signal from the display to the command stream parser during writes to this register. 0=Normal operation. 1=Force low during write of this register.
CRTC2_GUI_TRIG_VLINE (R)	31	0x0	This signal is active active when the raster is between the START and END. START <= raster <= END. The polarity is controlled as above. This signal goes to the command stream parser as a condition in the WAIT_UNTIL register. 0=Current line not between VLINE start and end. 1=Current line is between VLINE start and end, inclusive.

Trigger to GUI engine activated in certain vertical region of the display, when the raster is between START and END. Normally used to delay rendering operations until the raster has passed a specific point.

CRTC2_DEBUG - RW - 32 bits - [MMReg:0x31C]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_GUI_TRIG_BYPASS_EN	0	0x0	0=Don't bypass gui triggers generated by dispeng. 1=Bypass gui triggers generated by dispeng.

CRTC2_GUI_TRIG_VLINE_BYPASS	1	0x0	
CRTC2_GUI_TRIG_OFFSET_BYPASS	2	0x0	
CRTC2_GUI_TRIG_PITCH_ADD_BYPASS	3	0x0	

Controls for HW testing and debug of display systems. Should not be needed by software.

CRTC2_OFFSET - RW - 32 bits - [MMReg:0x324]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_OFFSET	26:0	0x0	Graphics surface origin offset in memory. Must be 8 byte aligned for 4bpp, 8bpp and 24bpp modes. Must be 16 byte aligned for all 16bpp modes. Must be 32 byte aligned for 32bpp mode. This is added to DISP_BASE_ADDR to get the full 32 bit address of the graphics surface. NOTE: Bits 0:2 of this field are hardwired to ZERO.
CRTC2_GUI_TRIG_OFFSET (R)	30	0x0	Indicates if visible buffer is last written, or still the previous one. This bit is read only. Goes high when an offset has been written but the corresponding buffer does not appear on screen yet. It goes low again when display starts for that address. 0=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written is being displayed 1=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written not yet displayed
CRTC2_OFFSET_LOCK	31	0x0	Prevents hardware from internally updating the following fields until cleared: CRTC2_OFFSET, CRTC2_TILE_LINE. It permits atomic update of CRTC2_OFFSET and CRTC2_TILE_LINE. Normal operation is with the lock in zero. 0=Unlock these regs 1=Lock'em

Secondary graphics visible surface origin.

CRTC2_OFFSET_CNTL - RW - 32 bits - [MMReg:0x328]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_TILE_LINE	3:0	0x0	When CRTC2_TILE_EN = 1, this field holds the 4 LSB of the line of the surface where CRTC2_OFFSET starts (the 'y' for line 0 of the display, or 'start line'). The display address generator needs to know this to determine the proper pitch to add at the end of each display line. This is normally 0, unless the display is in a virtual desktop mode with tiling enabled. For example, if the surface offset is zero and the display starts in line 3, CRTC2_TILE_LINE=0x3 CRTC2_OFFSET = 0xC0 Note that tiles are 8 lines high, but this register must contain ((start line) MOD 16) in order to do the checkerboarding correctly. Do not worry about what checkerboarding is, you shouldn't need to know.
CRTC2_TILE_EN	15	0x0	Graphics display tiling enable. 0=Display Surface uses linear addressing 1=Display surface uses tiled addressing
CRTC2_OFFSET_FLIP_CNTL	16	0x0	Selects position within the frame at which new CRTC2_OFFSET will be used. Should be normally zero. If set to one, a new offest will be taken at the end of the line instead of the end of the frame. 0=Use new CRTC2_OFFSET on vertical blank 1=Use new CRTC2_OFFSET on any horizontal blank. Note, this can cause the display to tear.
CRTC2_GUI_TRIG_OFFSET_LEFT_EN	28	0x1	Controls whether or not CRTC2_OFFSET affects the CRTC2_GUI_TRIG_OFFSET to the WAIT_UNTIL register. If enabled, then wait condition is set on writing CRTC2_OFFSET, and cleared only if display starts displaying from CRTC2_OFFSET. 0=Writing CRTC2_OFFSET (main/left) does not set CRTC2_GUI_TRIG_OFFSET 1=Writing CRTC2_OFFSET sets CRTC2_GUI_TRIG_OFFSET
CRTC2_GUI_TRIG_OFFSET *(mirror of CRTC2_OFFSET:CRTC2_GUI_TRIG_OFFSET)* (R)	30	0x0	Indicates if visible buffer is last written, or still the previous one. See CRTC2_OFFSET register. 0=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written is being displayed 1=Last CRTC_OFFSET or CRTC_OFFSET_RIGHT written not yet displayed

CRTC2_OFFSET_LOCK *(mirror of CRTC2_OFFSET:CRTC2_OFFSET_LOCK)*	31	0x0	Prevents hardware from internally updating certain fields until cleared. See CRTC2_OFFSET register. 0=Unlock these regs 1=Lock'em

Graphics Display Address Generator Control

CRTC2_PITCH - RW - 32 bits - [MMReg:0x32C]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_PITCH	10:0	0x0	Display line pitch in (pixels * 8). Note that for 24bpp the display uses pixels * 8 for the pitch, but the rendering engine uses bytes * 8 for the pitch. For tiled display this is the same pitch as used for the surface in the rendering engine (24bpp not supported for tiled). In tiled the pitch must be a multiple of 256 bytes (the tile width). So for 32bpp tiled, the CRTC2_PITCH must be a multiple of 64 pixels. Or 128 pixels for 16bpp, or 256 pixels for 8bpp. For stereo display mode, this is the left image pitch.

Graphics Display Address Pitch

CRTC2_DISPLAY_BASE_ADDR - RW - 32 bits - [MMReg:0x33C]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_DISPLAY_BASE_ADDR	31:0	0x0	Base address added to all graphics, cursor and icon requests to the internal memory controller. This should normally be set equal to MC_FB_LOCATION.MC_FB_START. It could be set to other places to allow display out of system (e.g. AGP) memory, but this is not recommended. This is a 4 Megabyte aligned base address. NOTE: Bits 0:21 of this field are hardwired to ZERO.

Secondary display graphics, cursor & icon base address

CRTC2_GEN_CNTL - RW - 32 bits - [MMReg:0x3F8]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_DBL_SCAN_EN	0	0x0	Double scan enable. Double scan only affects the calculation of display addresses by adding the CRTC_PITCH every second line (also applies to the hardware cursor pitch). Enabling double scan does not change the CRTC vertical programming or VSYNC timing. The overscan top & bottom are not affected and remain the number of lines programmed (i.e. not double). The hardware cursor programming remains in terms of physical lines (not logical lines). The cursor vertical position must begin on an even line number when in double scan. The cursor itself is limited to 64 physical lines in height, which means only 32 logical lines. This is because the cursor pitch is only added at the end of odd scan lines, but the CRTC vertical logic stops the cursor after 64 physical lines. 0=disable 1=enable
CRTC2_INTERLACE_EN	1	0x0	Interlace display mode enable. 0=Non-Interlace 1=Interlace
CRTC2_SYNC_TRISTATE	4	0x0	0=Normal 1=Tristate HSYNC and VSYNC outputs
CRTC2_HSYNC_TRISTATE	5	0x0	0=Normal HSYNC 1=Tristate HSYNC output
CRTC2_VSYNC_TRISTATE	6	0x0	0=Normal VSYNC 1=Tristate VSYNC output
CRT2_ON	7	0x0	0=CRT2 OFF - only CRT DAC used for CRT 1=CRT2 ON - TV DAC used for second CRT monitor (using CRTC2 path)

CRTC2_PIX_WIDTH	11:8	0x0	Display pixel width (actually depth). For 4bpp mode DAC_CNTL.DAC_4BPP_PIX_ORDER selects the order of nibbles within bytes. When R, G, or B components are only 4, 5 or 6 bits, DAC_CNTL.DAC_EXPAND_MODE selects how these components are expanded to 8 bits each for keying and palette lookup. When alpha values are 1 or 4 bits, they are expanded to 8 bits by dynamic expansion of the high order bits to the missing lower order bits. 0=Disable pixel clock for secondary CRTC 1=Reserved 2=8bpp Indexed 3=15bpp aRGB 1555 4=16bpp RGB 565 5=24bpp RGB 888 6=32bpp aRGB 8888 7=16bpp aRGB 4444 8=16bpp aIndex 88
CRTC2_ICON_EN	15	0x0	0=Disable Secondary Hardware Icon 1=Enable Secondary Hardware Icon
CRTC2_CUR_EN	16	0x0	Hardware cursor enable. This field is double buffered and locked with the CUR_LOCK register field. 0=Disable 1=Enable
CRTC2_CUR_MODE	22:20	0x0	Hardware cursor mode. For 2bpp mode, each line of cursor data is stored in memory as 64 bits of AND data followed by 64 bits or XOR data. For color AND/XOR mode, each pixel is stored sequentially in memory as 32bits each in aRGB8888 format with bit 31 of each DWord being the AND bit. For the color alpha modes the format is also 32bpp aRGB8888 with all 8 bits of the alpha being used. All HW cursor lines must be 64 pixels wide and all lines must be stored sequentially in memory. 0=Mono + 1 bit AND (2bpp), 64x64 1=Color 24bpp + 1 bit AND, 32hx64v 2=Color 24bpp + 8 bit aplha, premultiplied alpha, 32hx64v 3=Color 24bpp + 8 bit aplha, unmultiplied alpha, 32hx64v
CRTC2_DISPLAY_DIS	23	0x0	0=Enable 1=Blanked
CRTC2_EN	25	0x0	Enables CRT controller. When reset, the CRTC horizontal counter is set to zero, and the vertical counter is set to the current value of CRTC2_V_TOTAL_DISP.CRTC2_V_DISP. 0=Reset 1=Enable

CRTC2_DISP_REQ_EN_B	26	0x1	Enables display graphics requests to the memory controller. Affects only graphics and VGA text requests. Does not affect HW cursor, overlay or subpic. When setting this bit, CRTC2_EXT_CNTL.CRTC2_DISPLAY_DIS is also normally set to blank the screen. Active low. 0=Enable 1=Disable
CRTC2_C_SYNC_EN	27	0x0	0=Disable 1=Enable
CRTC2_HSYNC_DIS	28	0x0	0=Enable 1=Disable
CRTC2_VSYNC_DIS	29	0x0	0=Enable 1=Disable

Secondary CRTC general controls

CRTC2_STATUS - RW - 32 bits - [MMReg:0x3FC]			
Field Name	**Bits**	**Default**	**Description**
CRTC2_VBLANK_CUR (R)	0	0x0	Indicates if raster currently in vertical blank. 0=Not in vertical blank 1=In vertical blank
CRTC2_VBLANK_SAVE (R)	1	0x0	Clearable vertical blank indicator. Used by software to determine if still the same vertical blank as the last time polling (or interrupt) occured. 0=No vertical blank since last clear 1=Vertical Blank since last cleared
CRTC2_VBLANK_SAVE_CLEAR (W)	1	0x0	Used to clear CRTC2_VBLANK_SAVE. 0=No effect 1=Clear CRTC2_VBLANK_SAVE
CRTC2_VLINE_SYNC (R)	2	0x0	Indicates if the scan line is even or odd. 0=Even scan line 1=Odd scan line
CRTC2_FRAME (R)	3	0x0	Indicates if evan or odd frame currently displayed. 0=Even frame 1=Odd frame

Status bits to determine current state of the display.

CRTC_STATUS - RW - 32 bits - [IOReg,MMReg:0x5C]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_CUR (R)	0	0x0	Indicates if raster currently in vertical blank. 0=Not in vertical blank 1=In vertical blank
CRTC_VBLANK_SAVE (R)	1	0x0	Clearable vertical blank indicator. Used by software to determine if still the same vertical blank as the last time polling (or interrupt) occured. 0=No vertical blank since last clear 1=Vertical Blank since last cleared
CRTC_VBLANK_SAVE_CLEAR (W)	1	0x0	Used to clear CRTC_VBLANK_SAVE. 0=No effect 1=Clear CRTC_VBLANK_SAVE
CRTC_VLINE_SYNC (R)	2	0x0	Indicates if the scan line is even or odd. 0=Even scan line 1=Odd scan line
CRTC_FRAME (R)	3	0x0	Indicates if evan or odd frame currently displayed. 0=Even frame 1=Odd frame

Graphics & cursor base address

2.23 DDC Registers

GPIO_VGA_DDC - RW - 32 bits - [IOReg,MMReg:0x60]			
Field Name	**Bits**	**Default**	**Description**
VGA_DCC_DATA_OUTPUT	0	0x0	Ouput data bit if output enabled by VGA_DDC_DATA_EN. 0=Drive VGA_DDC_DATA low, if enabled. 1=Drive VGA_DDC_DATA high, if enabled.
VGA_DDC_CLK_OUTPUT	1	0x0	Ouput clock bit if output enabled by VGA_DDC_CLK_EN. 0=Drive VGA_DDC_CLK low, if enabled. 1=Drive VGA_DDC_CLK high, if enabled.
VGA_DDC_DATA_INPUT (R)	8	0x0	Data bit input. 0=VGA_DDC_DATA pin input is low. 1=VGA_DDC_DATA pin input is high.
VGA_DDC_CLK_INPUT (R)	9	0x0	Clock bit input. 0=VGA_DDC_CLK pin input is low. 1=VGA_DDC_CLK pin input is high.
VGA_DDC_DATA_OUT_EN	16	0x0	Data pin output enable. 0=VGA_DDC_DATA output disabled. 1=VGA_DDC_DATA output enabled.
VGA_DDC_CLK_OUT_EN	17	0x0	Clock pin output enable. 0=VGA_DDC_CLK output disabled. 1=VGA_DDC_CLK output enabled.

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

GPIO_DVI_DDC - RW - 32 bits - [IOReg,MMReg:0x64]			
Field Name	**Bits**	**Default**	**Description**
DVI_DDC_DATA_OUTPUT	0	0x0	Ouput data bit if output enabled by DVI_DDC_DATA_EN. 0=Drive DVI_DDC_DATA low, if enabled. 1=Drive DVI_DDC_DATA high, if enabled.
DVI_DDC_CLK_OUTPUT	1	0x0	Ouput clock bit if output enabled by DVI_DDC_CLK_EN. 0=Drive DVI_DDC_CLK low, if enabled. 1=Drive DVI_DDC_CLK high, if enabled.
DVI_DDC_DATA_INPUT (R)	8	0x0	Data bit input. 0=DVI_DDC_DATA pin input is low. 1=DVI_DDC_DATA pin input is high.

DVI_DDC_CLK_INPUT (R)	9	0x0	Clock bit input. 0=DVI_DDC_CLK pin input is low. 1=DVI_DDC_CLK pin input is high.
DVI_DDC_DATA_OUT_EN	16	0x0	Data pin output enable. 0=DVI_DDC_DATA output disabled. 1=DVI_DDC_DATA output enabled.
DVI_DDC_CLK_OUT_EN	17	0x0	Clock pin output enable. 0=DVI_DDC_CLK output disabled. 1=DVI_DDC_CLK output enabled.
SW_WANTS_TO_USE_DVI_I2C *(mirror of DVI_I2C_CNTL_0:SW_WANTS_TO_USE_DVI_I2C)* (W)	20	0x0	0=Normal 1=SW requests to use DVI I2C interface
SW_CAN_USE_DVI_I2C *(mirror of DVI_I2C_CNTL_0:SW_CAN_USE_DVI_I2C)* (R)	20	0x0	0=DVI I2C interface not available 1=SW has control of the DVI I2C interface
SW_DONE_USING_DVI_I2C *(mirror of DVI_I2C_CNTL_0:SW_DONE_USING_DVI_I2C)* (W)	21	0x0	0=if SW has control of DVI I2C, free to use it 1=indicate SW is done using DVI I2C i/f
HDCP_NEEDS_DVI_I2C *(mirror of DVI_I2C_CNTL_0:HW_NEEDS_DVI_I2C)* (R)	21	0x0	0=no request by H/W for use of DVI I2C 1=H/W HDCP requests use of the DVI I2C i/f
ABORT_HDCP_DVI_I2C *(mirror of DVI_I2C_CNTL_0:ABORT_HW_DVI_I2C)* (W)	22	0x0	0=normal operation of H/W using DVI I2C 1=abort current H/W HDCP use of DVI I2C interface
HW_USING_DVI_I2C *(mirror of DVI_I2C_CNTL_0:HW_USING_DVI_I2C)* (R)	22	0x0	0=DVI I2C i/f not in use by H/W 1=HDCP H/W currently using DVI I2C interface

Control and read of DDC clock and data lines for digital connector.

GPIO_MONID - RW - 32 bits - [IOReg,MMReg:0x68]			
Field Name	**Bits**	**Default**	**Description**
GPIO_MONID_0_OUTPUT	0	0x0	Output data for MONID(0) pin. 0=Drive MONID(0) low, if enabled. 1=Drive MONID(0) high, if enabled.

GPIO_MONID_1_OUTPUT	1	0x0	Output data for MONID(1) pin. 0=Drive MONID(0) low, if enabled. 1=Drive MONID(1) high, if enabled.
GPIO_MONID_0_INPUT (R)	8	0x0	Input data from MONID(0) pin. 0=MONID(0) pin input is low. 1=MONID(0) pin input is high.
GPIO_MONID_1_INPUT (R)	9	0x0	Input data from MONID(1) pin. 0=MONID(1) pin input is low. 1=MONID(1) pin input is high.
GPIO_MONID_0_OUT_EN	16	0x0	Output enable for MONID(0) pin. 0=MONID(0) output disabled. 1=MONID(0) output enabled.
GPIO_MONID_1_OUT_EN	17	0x0	Output enable for MONID(1) pin. 0=MONID(0) output disabled. 1=MONID(1) output enabled.

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

GPIO_CRT2_DDC - RW - 32 bits - [IOReg,MMReg:0x6C]			
Field Name	**Bits**	**Default**	**Description**
CRT2_DDC_DATA_OUTPUT	0	0x0	0=Drive CRT2_DDC_DATA low, if enabled. 1=Drive CRT2_DDC_DATA high, if enabled.
CRT2_DDC_CLK_OUTPUT	1	0x0	0=Drive CRT2_DDC_CLK low, if enabled. 1=Drive CRT2_DDC_CLK high, if enabled.
CRT2_DDC_DATA_INPUT (R)	8	0x0	0=CRT2_DDC_DATA pin input is low. 1=CRT2_DDC_DATA pin input is high.
CRT2_DDC_CLK_INPUT (R)	9	0x0	0=CRT2_DDC_CLK pin input is low. 1=CRT2_DDC_CLK pin input is high.
CRT2_DDC_DATA_OUT_EN	16	0x0	0=CRT2_DDC_DATA output disabled. 1=CRT2_DDC_DATA output enabled.
CRT2_DDC_CLK_OUT_EN	17	0x0	0=CRT2_DDC_CLK output disabled. 1=CRT2_DDC_CLK output enabled.

No description available for this register.

2.24 overlay Registers

OV0_Y_X_START - RW - 32 bits - [MMReg:0x400]			
Field Name	**Bits**	**Default**	**Description**
OV0_X_START	12:0	0x0	Overlay X starting coordinate relative to ACTIVE screen (0,0)
OV0_Y_START	28:16	0x0	Overlay Y starting coordinate relative to ACTIVE screen (0,0)

X/Y coords relative to active display in pixels

OV0_Y_X_END - RW - 32 bits - [MMReg:0x404]			
Field Name	**Bits**	**Default**	**Description**
OV0_X_END	12:0	0x0	Overlay X ending coordinate relative to ACTIVE screen (0,0)
OV0_Y_END	28:16	0x0	Overlay X ending coordinate relative to ACTIVE screen (0,0)

X/Y ending coordinates (inclusive) relative to active display in pixels

OV0_PIPELINE_CNTL - RW - 32 bits - [MMReg:0x408]			
Field Name	**Bits**	**Default**	**Description**
OV0_DISP_PIPE_DELAY	3:0	0x8	Debug field used to adjust overlay window position. Leave at default unless hardware problem requires adjustment as instructed.

Debug adjustments for overlay window control

OV0_REG_LOAD_CNTL - RW - 32 bits - [MMReg:0x410]			
Field Name	**Bits**	**Default**	**Description**
OV0_LOCK	0	0x0	If this bit is set, then the start of Display VBlank event that usually updates double buffered registers is ignored. Use this bit to autonomously update scaler and subpicture registers. To use, set this bit, and then poll OV0_LOCK_READBACK until the lock takes effect. In almost all situations the lock should take effect before the first read; however, it is theoretically possible that the lock will not take quickly if various clocks in the chip are programmed to very low frequencies (for power saving for example). Once the lock takes, write registers to update the overlay. Then write the lock bit to zero. The new settings will take effect at the next VBlank rising edge (provided that LOCK is left unset). 0=NOT LOCKED 1=LOCKED
OV0_VBLANK_DURING_LOCK (R)	1	0x0	This bit is updated only when OV0_LOCK transitions from 1 to 0. If VBlank occured while OV0_LOCK was 1, then this bit will be updated to the value one. Otherwise it is updated to the value zero. 0=DID NOT HAPPEN 1=DID HAPPEN
OV0_STALL_GUI_UNTIL_FLIP	2	0x0	The intent here is for the overlay to be able to tell the GUI that it is using the surface that the GUI wants to render to. The overlay will send an 'OV0_SURFACE_IS_FREE' signal to the GUI. It will make this signal go low when there is a danger of front buffer overwrite as determined by software. If software wants to stall the GUI, then it will set OV0_STALL_GUI_UNTIL_FLIP when it locks, updates, and unlocks overlay and subpicture registers. OV0_SURFACE_IS_FREE will go low at unlock and then high during VBlank (when the hardware double buffering flips the registers). The behavior of OV0_SURFACE_IS_FREE is undefined if OV0_STALL_GUI_UNTIL_FLIP is written to when the lock bit is not set. OV0_SURFACE_IS_FREE is not an event signal. If it is low the WaitUntilEvent command must stall the GUI until it is high. It does not wait until the signal transitions from low to high. (i.e. If it is already high, there is no stall). 0=DO NOT STALL 1=DO STALL

OV0_LOCK_READBACK (R)	3	0x0	See OV0_LOCK. This bit indicates whether the lock took effect. 0=NOT LOCKED 1=LOCKED
OV0_FLIP_READBACK (R)	4	0x0	Status indication of whether last submitted overlay flip has occured yet. 0=Flip not yet occured. 1=Flip has occured.

Overlay register load control

OV0_SCALE_CNTL - RW - 32 bits - [MMReg:0x420]			
Field Name	**Bits**	**Default**	**Description**
OV0_NO_READ_BEHIND_SCAN	1	0x0	Recommend set to zero. If set, the data for the next display line is not written to the line buffer until the current line has been read.
OV0_HORZ_PICK_NEAREST	2	0x0	Recommend set to zero. If source keying is absolutely needed, then we may need to drop the interpolated scale in order to make the source keyer work. 0=BLEND 1=FIX H_ALPHA = 0
OV0_VERT_PICK_NEAREST	3	0x0	Recommend set to zero. If source keying is absolutely needed, then we may need to drop the interpolated scale in order to make the source keyer work. 0=BLEND 1=FIX V_ALPHA = 0
OV0_SIGNED_UV	4	0x0	Support the signed UV source format needed by Apple. 0=UNSIGNED 1=SIGNED
OV0_GAMMA_SEL	7:5	0x0	Select gamma correction factor: (default=0) 0 = Brightness Enable 1 = Gamma 2.2 2 = Gamma 1.8 3 = Gamma 1.4 0=Linear, G=1.0 1=G=0.85 2=G=1.1 3=G=1.2 4=G=1.45 5=G=1.7 6=G=2.2 7=G=2.5

OV0_SURFACE_FORMAT	11:8	0x0	Scale source pixel format from memory: 0-2 = (reserved) 3 = 15 bpp aRGB 1555 4 = 16 bpp RGB 565 5 = (reserved) 6 = 32 bpp aRGB 8888 7-8 = (reserved) 9 = YUV 9 10 = YUV 12 11 = VYUY422 12 = YVYU422 13-15 = (reserved) 3=15BPP ARGB1555 4=16BPP RGB565 6=32BPP ARGB8888 9=Three plane YUV9 10=Three plane YUV12 11=Packed VYUY422 12=Packed YVYU422 13=Two plane YUV12
OV0_ADAPTIVE_DEINT	12	0x0	0=Adaptive de-interlacing is off. 1=Adaptive de-interlacing enabled.
OV0_CRTC_SEL	14	0x0	0=Use overlay/scaler on primary CRTC display path 1=Use overlay/scaler on secondary CRTC2 display path
OV0_BURST_PER_PLANE	22:16	0x7f	

OV0_DOUBLE_BUFFER_REGS	24	0x0	Recommend set to 1. Turns on double buffering so that many registers can be updated autonomously in the VBlank. Currently the following fields are double buffered but the list is subject to change as we attempt to reduce the gate count. OV0_V_INC, OV0_P1_V_ACCUM_INIT, OV0_P23_V_ACCUM_INIT, OV0_P1_BLNK_LN_AT_TOP_M1, OV0_P1_ACTIVE_LINES_M1, OV0_P23_BLNK_LN_AT_TOP_M1, OV0_P23_ACTIVE_LINES_M1, OV0_P1_OCTWORDS_PER_LINE_M1, OV0_P2_OCTWORDS_PER_LINE_M1, OV0_P3_OCTWORDS_PER_LINE_M1, OV0_SMART_SWITCH, OV0_BURST_PER_PLANE, OV0_SOFT_EOF_TOGGLE, OV0_DEINT_PAT, OV0_DEINT_PAT_LEN_M1, OV0_P1_H_INC, OV0_P23_H_INC, OV0_P1_H_STEP_BY, OV0_P23_H_STEP_BY, OV0_P1_H_ACCUM_INIT, OV0_PRESHIFT_P1_TO, OV0_P23_H_ACCUM_INIT, OV0_PRESHIFT_P23_TO, OV0_P1_X_START, OV0_P1_X_END, OV0_P2_X_START, OV0_P2_X_END, OV0_P3_X_START, OV0_P3_X_END, OV0_P1_MAX_LN_IN_PER_LN_OUT, OV0_P23_MAX_LN_IN_PER_LN_OUT. 0=OFF 1=ON
OV0_BANDWIDTH (R)	26	0x0	0=NORMAL 1=BANDWIDTH LIMIT/ACK
OV0_LIN_TRANS_BYPASS	28	0x0	0=DO NOT BYPASS COLOUR CONVERSION 1=BYPASS COLOUR CONVERSION
OV0_INT_EMU	29	0x0	Enables experimental deinterlacing hardware which should make displayed video look sharper than 'bobbed' video, but without the feathering artifacts associated with 'weaved' video. It also makes colours richer. 0=OFF 1=ON
OV0_OVERLAY_EN	30	0x0	0 = Overlay Disable (default=0) 1 = Overlay Enable 0=DISABLE 1=ENABLE
OV0_SOFT_RESET	31	0x1	Resets the scaler. 0=ENABLE 1=RESET

General scaler control

OV0_V_INC - RW - 32 bits - [MMReg:0x424]			
Field Name	**Bits**	**Default**	**Description**
OV0_V_INC	25:8	0x0	Vertical accumulator increment [17:0]. This is a 6.12 fixed point number. It is used to control the vertical scaling ratio.

Vertical accumulator increment

OV0_P1_V_ACCUM_INIT - RW - 32 bits - [MMReg:0x428]			
Field Name	**Bits**	**Default**	**Description**
OV0_P1_MAX_LN_IN_PER_LN_OUT	1:0	0x1	This field supports a new feature which is the ability to read in and blend more than one line for every line output. Using it greatly improves image quality when downscaling vertically. However, memory bandwidth limits the number of lines that can be read in. Do not program the scaler to read in more lines than there is bandwidth to support. The scaler document provides an explanation on determining the available bandwith.
OV0_P1_V_ACCUM_INIT	25:15	0x0	This is a 6.5 fixed point number. The integer protion indicates how many lines to fetch and cycle into (and possibly through) through the line buffers in preparation for generating the first display line. The fractional portion indicates the blend of source lines the four tap/two tap filter kernal that is used to generate the first display line.

This register is needed to accurately vertically position the source video in the overlay display window, and also to accurately position it relative to the subpicture image.

OV0_P23_V_ACCUM_INIT - RW - 32 bits - [MMReg:0x42C]			
Field Name	**Bits**	**Default**	**Description**
OV0_P23_MAX_LN_IN_PER_LN_OUT	1:0	0x1	This field supports a new feature which is the ability to read in and blend more than one line for every line output. Using it greatly improves image quality when downscaling vertically. However, memory bandwidth limits the number of lines that can be read in. Do not program the scaler to read in more lines than there is bandwidth to support. The scaler document provides an explanation on determining the available bandwith.

OV0_P23_V_ACCUM_INIT	24:15	0x0	This is a 6.5 fixed point number. The integer protion indicates how many lines (planer U and V lines) to fetch and cycle into (and possibly through) through the line buffers in preparation for generating the first display line. The fractional portion indicates the blend of source lines the four tap/two tap filter kernal that is used to generate the first display line.

Please refer to P1_V_ACCUM_INIT. This register is used for the UV components in Planer modes.

OV0_P1_BLANK_LINES_AT_TOP - RW - 32 bits - [MMReg:0x430]			
Field Name	**Bits**	**Default**	**Description**
OV0_P1_BLNK_LN_AT_TOP_M1	11:0	0xfff	This is a signed value. Default it to -1 and don't exceed the maximum positive value.
OV0_P1_ACTIVE_LINES_M1	27:16	0x0	Program this with the surface height minus one. After the hardware scales and displays the blank lines at top and the active lines, it will continue to scale and display blank lines until it reaches the end of the overlay window.

Used for DVD letterboxing (when a wide movie is displayed with black borders at the top and bottom). The problem is that software can't provide the black boarders using the primary display because subpicture information can be displayed in the black regions. It also is wasteful to allocate extra off screen memory in the frame buffer and fill it with 'black' pixels. Thus use these fields to create the black borders for DVD letterboxing.

OV0_P23_BLANK_LINES_AT_TOP - RW - 32 bits - [MMReg:0x434]			
Field Name	**Bits**	**Default**	**Description**
OV0_P23_BLNK_LN_AT_TOP_M1	10:0	0x7ff	This is a signed value. Default it to -1 and don't exceed the maximum positive value.
OV0_P23_ACTIVE_LINES_M1	26:16	0x0	Program this with the surface height minus one. After the hardware scales and displays the blank lines at top and the active lines, it will continue to scale and display blank lines until it reaches the end of the overlay window.

Please refer to P1_BLANK_LINES_AT_TOP. This register is used for the UV components in Planer modes.

OV0_BASE_ADDR - RW - 32 bits - [MMReg:0x43C]			
Field Name	**Bits**	**Default**	**Description**
OV0_BASE_ADDR	31:0	0x0	This base is added to all overlay address requests to the internal memory controller. Setting this register equal to MC_FB_LOCATION.MC_FB_START positions all overlay surfaces in the frame buffer. Alternatively setting it to MC_AGP_LOCATION.MC_AGP_START positions overlay surfaces in AGP memory. The use of the overlay from AGP is supported, but not recommended. This base address must be a 4 Megabyte aligned value. NOTE: Bits 0:21 of this field are hardwired to ZERO.

Overlay base address

OV0_VID_BUF0_BASE_ADRS - RW - 32 bits - [MMReg:0x440]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_BUF_PITCH_SEL	0	0x0	Associates either OV0_VID_BUF_PITCH0_VALUE register or OV0_VID_BUF_PITCH1_VALUE register with this surface or plane of a planer surface. 0=SELECT PITCH 0 1=SELECT PITCH 1
RESERVED_BIT1	1	0x0	reserved for tiling info. No overlay tiling in this ASIC.
OV0_VID_BUF_BASE_ADRS	26:4	0x0	BASE_ADRS is actually a poor choice of name. This field points to the octaword that contains the top left corner pixel of the region of the surface that you wish to display.
RESERVED_BIT31_28	31:28	0x0	reserved for tiling info. No overlay tiling in this ASIC.

A surface desriptor register that include the base address (or base offset) of a surface, and some attributes for pitch selection aand tiled surface support.

Double buffering (for autonomous overlay updates), or multi-buffering can be achieved by swtching between any of the six base address registers using the OV0_AUTOFLIP_CNTL register. (Note: for planer surfaces the six registers are interpreted by hardware as two sets of three registers. Thus only OV0_VID_BUF0 and OV0_VID_BUF3 are valid selections.)

OV0_VID_BUF0 can describe any non-planer surface, or it can be used to describe the Y plane of a planer surface.

OV0_VID_BUF1_BASE_ADRS - RW - 32 bits - [MMReg:0x444]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_BUF_PITCH_SEL	0	0x0	Same as in OV0_VID_BUF0_BASE_ADRS. 0=SELECT PITCH 0 1=SELECT PITCH 1
RESERVED_BIT1	1	0x0	reserved for tiling info. No overlay tiling in this ASIC.
OV0_VID_BUF_BASE_ADRS	26:4	0x0	Same as in OV0_VID_BUF0_BASE_ADRS.
RESERVED_BIT31_28	31:28	0x0	reserved for tiling info. No overlay tiling in this ASIC.

OV0_VID_BUF1 can describe any non-planer surface, or it can be used to describe the U plane of a planer surface.

OV0_VID_BUF2_BASE_ADRS - RW - 32 bits - [MMReg:0x448]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_BUF_PITCH_SEL	0	0x0	Same as in OV0_VID_BUF0_BASE_ADRS. 0=SELECT PITCH 0 1=SELECT PITCH 1
RESERVED_BIT1	1	0x0	reserved for tiling info. No overlay tiling in this ASIC.
OV0_VID_BUF_BASE_ADRS	26:4	0x0	Same as in OV0_VID_BUF0_BASE_ADRS.
RESERVED_BIT31_28	31:28	0x0	reserved for tiling info. No overlay tiling in this ASIC.

OV0_VID_BUF2 can describe any non-planer surface, or it can be used to describe the V plane of a planer surface.

OV0_VID_BUF3_BASE_ADRS - RW - 32 bits - [MMReg:0x44C]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_BUF_PITCH_SEL	0	0x0	Same as in OV0_VID_BUF0_BASE_ADRS. 0=SELECT PITCH 0 1=SELECT PITCH 1
RESERVED_BIT1	1	0x0	reserved for tiling info. No overlay tiling in this ASIC.
OV0_VID_BUF_BASE_ADRS	26:4	0x0	Same as in OV0_VID_BUF0_BASE_ADRS.
RESERVED_BIT31_28	31:28	0x0	reserved for tiling info. No overlay tiling in this ASIC.

OV0_VID_BUF3 can describe any non-planer surface, or it can be used to describe the Y plane of a planer surface.

OV0_VID_BUF4_BASE_ADRS - RW - 32 bits - [MMReg:0x450]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_BUF_PITCH_SEL	0	0x0	Same as in OV0_VID_BUF0_BASE_ADRS. 0=SELECT PITCH 0 1=SELECT PITCH 1
RESERVED_BIT1	1	0x0	reserved for tiling info. No overlay tiling in this ASIC.
OV0_VID_BUF_BASE_ADRS	26:4	0x0	Same as in OV0_VID_BUF0_BASE_ADRS.
RESERVED_BIT31_28	31:28	0x0	reserved for tiling info. No overlay tiling in this ASIC.

OV0_VID_BUF4 can describe any non-planer surface, or it can be used to describe the U plane of a planer surface.

OV0_VID_BUF5_BASE_ADRS - RW - 32 bits - [MMReg:0x454]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_BUF_PITCH_SEL	0	0x0	Same as in OV0_VID_BUF0_BASE_ADRS. 0=SELECT PITCH 0 1=SELECT PITCH 1
RESERVED_BIT1	1	0x0	reserved for tiling info. No overlay tiling in this ASIC.
OV0_VID_BUF_BASE_ADRS	26:4	0x0	Same as in OV0_VID_BUF0_BASE_ADRS.
RESERVED_BIT31_28	31:28	0x0	reserved for tiling info. No overlay tiling in this ASIC.

OV0_VID_BUF5 can describe any non-planer surface, or it can be used to describe the V plane of a planer surface.

OV0_VID_BUF_PITCH0_VALUE - RW - 32 bits - [MMReg:0x460]			
Field Name	**Bits**	**Default**	**Description**
OV0_PITCH_VALUE	19:4	0x0	For linear (non-tiled) surfaces, the pitch value is the pitch of the surface, or in the case of planer modes, the pitch of a plane in a surface. For tiled modes, the pitch value is not the pitch. It is a number that the hardware needs to get from the beginning of last line that it read in the previous row of tiles to the beginning first line that it must read in the next row of tiles. To understand this concept, you need a solid understanding of how tiled surfaces are organized in memory. (there is code in the ov0setup.cpp sample code that does the neccesary calculations).

OV0_PITCH_SKIP_LINES	27:26	0x0	Used only if the surface (or plane in planer surface) is tiled. Ignored otherwise. When downscaling, or when picking off the even or odd lines of an MPEG frame, it is sometimes desireable to 'double the pitch'. However, unlike linear surfaces, doubling the pitch in a tiled surface doesn't have the desired effect. Therefore, hardware provides a mechanism for skipping lines. Using this filed will require that the OV0_PITCH?_VALUE field be changed as well. 0=USE EVERY LINE 1=USE EVERY 2nd LINE 2=USE EVERY 4th LINE 3=USE EVERY 8th LINE
OV0_PITCH_IN_TILES_LSBS	31:28	0x0	Not used. No overlay tiling in this ASIC.

Fields needed for advancing the hardware's line start address pointer from the beginning of one line to the beginning of the next, and for enabling correct checkerboarding in tiled modes.

OV0_VID_BUF_PITCH1_VALUE - RW - 32 bits - [MMReg:0x464]			
Field Name	**Bits**	**Default**	**Description**
OV0_PITCH_VALUE	19:4	0x0	See OV0_VID_BUF_PITCH0_VALUE.
OV0_PITCH_SKIP_LINES	27:26	0x0	See OV0_VID_BUF_PITCH0_VALUE. 0=USE EVERY LINE 1=USE EVERY 2nd LINE 2=USE EVERY 4th LINE 3=USE EVERY 8th LINE
OV0_PITCH_IN_TILES_LSBS	31:28	0x0	Not used. No overlay tiling in this ASIC.

A second set of pitch fields that can be used to implement double buffering when changing the pitch, or that can be used to have different pitches for different planes in planer modes (YUV9 and YUV12). Selection of Pitch0/Pitch1 is controlled by the OV0_VID_BUF?_PITCH_SEL field in the selected OV0_VID_BUF?_BASE_ADRS register.

OV0_AUTO_FLIP_CNTRL - RW - 32 bits - [MMReg:0x470]			
Field Name	**Bits**	**Default**	**Description**
OV0_SOFT_BUF_NUM	2:0	0x0	A pointer to one of the six base address registers. For non-planer modes valid values are from 0 to 5. For planer modes valid values are 0 and 3.
OV0_SOFT_REPEAT_FIELD	3	0x0	0=LAST FIELD NOT REPEAT 1=LAST FIELD IS REPEAT

OV0_SOFT_BUF_ODD	4	0x0	For non-interlaced video (e.g. MPEG frames) this field is don't care, but the SHIFT_EVEN_DOWN and SHIFT_ODD_DOWN register fields should be zero. For interlaced video provided by a software application, this field should indicate if the video field is even or odd. If This value is ignored until the internal version of the _EOF_TOGGLE signal changes state. 0=EVEN 1=ODD
OV0_IGNORE_REPEAT_FIELD	5	0x0	If a hardware video capture port is selected, and if the video input stream has embedded repeat field flags, then these can be used to reset the deinterlacing pattern. Specifically, the current pattern will be used to select how the field that just arrived and it's two predeccessors are displayed. The next time a field arrives, deinterlacing pattern zero will be used. Setting this bit will cause the repeat field flags to be ignored. If VID_PORT_SELECT is 2 (software), then the hardware assumes that every incoming field has a REPEAT_FIELD attribute. Use this bit to disable that attibute on a field by field basis. 0=DON'T IGNORE 1=DO IGNORE
OV0_SOFT_EOF_TOGGLE	6	0x0	After writting SOFT_BUF_NUM and SOFT_BUF_ODD, then change the state of the EOF_TOGGLE bit. The EOF_TOGGLE signal is double buffered, so your change won't happen to the internal version of EOF_TOGGLE (and thus won't be used by hardware) until the start of VBlank, and then only if the LOCK bit is low. However when the internal version does change state, the hardware becomes aware of the new field/frame of video that you describe using SOFT_BUF_NUM and SOFT_BUF_ODD.
OV0_VID_PORT_SELECT	9:8	0x0	Selects one of the two capture engines or a software application as the provider of video. 0=PORT 0 1=reserved 2=SOFTWARE
OV0_P1_FIRST_LINE_EVEN	16	0x0	If you are weaving two fields together, then you must indicate if the first line that the scaler will fetch is in the field labeld even or the field labeld odd. 0=IS ODD 1=IS EVEN
OV0_SHIFT_EVEN_DOWN	18	0x0	If 'bobbing' video (Called 'Run' Deinterlacing) then this bit allows you to shift the even fields down.
OV0_SHIFT_ODD_DOWN	19	0x0	If 'bobbing' video (Called 'Run' Deinterlacing) then this bit allows you to shift the odd fields down.

OV0_FIELD_POL_SOURCE	23	0x0	0=Video field polarity taken from next source field. Recommended for bob. 1=Video field polarity taken from current source field. Recommended for weave.

As a video capture port or software application provides new fields or frames of video, information about where these new fields/frames, what they are, and how they should be displayed is passed to the scaler using the AUTO_FLIP_CNTRL register.

OV0_SOFT_BUF_NUM, OV0_SOFT_BUF_ODD, and OV0_IGNORE_REPEAT_FIELD are a software application's way of describing new video fields or frames to hardware. These values are ignored until the internal (double buffered) version of the _EOF_TOGGLE signal changes state.

If a new video field or frame is created, indicate where it is by writing the address into an unused BASE ADDRESS REGISTER. (If you have already set up a circular buffer then this may already be done). Then write the number of the base address register in OV0_SOFT_BUF_NUM. Also set it's attributes using OV0_SOFT_BUF_ODD and OV0_IGNORE_REPEAT_FIELD. Then change the value of SOFT_EOF_TOGGLE to submit the description of the new field to hardware.

The hardware will display some 'combination' of this field and the two previous fields. ('Weave' most recent with next most recent, or 'Weave' the second and third most recent, 'Bob' most recent, or 'Bob' the second most recent)

OV0_DEINTERLACE_PATTERN - RW - 32 bits - [MMReg:0x474]

Field Name	Bits	Default	Description
OV0_DEINT_PAT	19:0	0x0	Ten two bit values. Values are enumerated as follows: 0) weave the two most recently submitted fields together and display them, 1) weave the second two most recently submitted fields together and display them, 2) display only the most recently submitted field (using 'bob' deinterlacing), 3) display only the second most recently submitted field (using 'bob' deinterlacing), The scaler documentation provides example patterns of two bit values that can be programmed into this field. This field is double buffered. A two bit value is sampled from the position indicated by the OV0_DEINT_PAT_PNTR at the begining of a display frame after a new video field is submitted. This two bit value remains in use untill a new field is submitted.
OV0_DEINT_PAT_PNTR (R)	27:24	0x0	Read this to find out which two bit value in the pattern the video is currently using.
OV0_DEINT_PAT_LEN_M1	31:28	0x0	Set this to 9 if you want to cycle through all ten two bit values, else set it to a lower value.

The scaler remembers the last three fields of video that were provided. There are 10 two bit values in the OV0_DEINETRLACE_PATTERN register. Each two bit value selects one of four ways that these fields should be displayed.

OV0_SUBMIT_HISTORY - R - 32 bits - [MMReg:0x478]			
Field Name	**Bits**	**Default**	**Description**
OV0_NEXT_BUF_NUM	2:0	0x0	
OV0_NEXT_IS_ODD	4	0x0	0=EVEN 1=ODD
OV0_CURR_BUF_NUM	10:8	0x0	
OV0_CURR_IS_ODD	12	0x0	0=EVEN 1=ODD
OV0_PREV_BUF_NUM	18:16	0x0	
OV0_PREV_IS_ODD	20	0x0	0=EVEN 1=ODD

No description available for this register.

OV0_H_INC - RW - 32 bits - [MMReg:0x480]			
Field Name	**Bits**	**Default**	**Description**
OV0_P1_H_INC	13:0	0x0	Horizontal scale increment (1/Scale Ratio) for Y, R, G, and B. It is a 2.12 fixed point number.
OV0_P23_H_INC	29:16	0x0	Horizontal scale increment (1/Scale Ratio) for U and V (Actually Cb and Cr). It is a 2.12 fixed point number.

This register indicates how much to scale the Y, R, G, B (P1) or UV (P23) in the horizontal direction. Note that to improve quality when downscaling you should decimate the data first using the 'STEP_BY' register fields. The decimated data is then scaled by the amount that you indicate using P?_H_INC. The CalcH.cpp function provides the decision making code required to do this.

OV0_STEP_BY - RW - 32 bits - [MMReg:0x484]			
Field Name	**Bits**	**Default**	**Description**
OV0_P1_H_STEP_BY	2:0	0x0	P1 refers to the Y in YUV, to G in RGB32, and to RGB in RGB 15/16. In RGB15/16 the value zero (four tap vertical filtering) is not supported. 0=STEP BY 1, 4TAP VBLEND 1=STEP BY 1, 2TAP VBLEND 2=STEP BY 2, 2TAP VBLEND 3=STEP BY 4 or 3/5, 2TAP VBLEND 4=STEP BY 8 or 7/9 or 6/10, 2TAP VBLEND 5=STEP BY 15/17 or 14/18, 2TAP VBLEND

OV0_P1_PREDWNSC_RATIO	4	0x0	Enables horizontal pre-downscaling for use when horizontal downscale ratio of P1 data is less than 2:1. 0=No predownscale, i.e. 1:1 1=Predownscale 2:1
OV0_P23_H_STEP_BY	10:8	0x0	P23 refers to the UV in YUV, and to RB in RGB32. It is not used in RGB15/16 0=PLANER UV STEP BY 1, 4TAP VBLEND 1=PLANER UV STEP BY 1, 2TAP VBLEND 2=PLANER UV STEP BY 2, 2TAP VBLEND 3=PLANER UV STEP BY 4, 2TAP VBLEND 4=PLANER UV STEP BY 8, 2TAP VBLEND 5=PLANER UV STEP BY 14/18, 2TAP VBLEND
OV0_P23_PREDWNSC_RATIO	12	0x0	Enables horizontal pre-downscaling for use when horizontal downscale ratio of P23 data is less than 2:1. In modes where P23 data is already sub-sampled horizontally, this would only be used when the horizontal scale ratio goes below 4:1. 0=No predownscale, i.e. 1:1 1=Predownscale 2:1

Horizontal scaling is performed via a combination of horizontal blending and horizontal decimation. There is a limit to how many source pixels the vertical scalers can provide to the horizontal scalers per output pixel. If H_STEP_BY is zero, then the vertical filters perform four tap vertical filtering. If H_STEP_BY is greater than zero, then the the vertical scalers reconfigure themselves so that they provide twice as many pixels per clock, but perform only linear two tap vertical filtering to generate those pixels. If H_STEP_BY is 2, then the vertical filters fetch only the pixels with and even X coordinate relative to the octword alligned surface base offset. If H_STEP_BY is 3, then every 4th pixel is fetched, etc. For ease of implementation reasons, at higher H_STEP_BY settings, the spacing between may not be an exact power of two. For example when H_STEP_BY is 4, the hardware may step by 7,9,7,9 rather than stepping by 8,8,8,8.

The code to program this field is provided in a function called CalcH.cpp

OV0_P1_H_ACCUM_INIT - RW - 32 bits - [MMReg:0x488]			
Field Name	**Bits**	**Default**	**Description**
OV0_P1_H_ACCUM_INIT	19:15	0x0	Indicates the blend of source pixels you want in the four tap filter kernal to provide to generate the first display pixel. The ov0setup.cpp program shows how to calculate the correct value.
OV0_PRESHIFT_P1_TO	31:28	0x2	Indicates how many source pixels you want to preshift into the horizontal Y four tap filter to generate the first display pixel. Please refer to the documentation for sample code on how to program this field. Typically this value is two or three.

This register is needed to accurately horizontally position the source video in the overlay display window, and also to accurately position it relative to the subpicture image.

OV0_P23_H_ACCUM_INIT - RW - 32 bits - [MMReg:0x48C]			
Field Name	**Bits**	**Default**	**Description**
OV0_P23_H_ACCUM_INIT	19:15	0x0	Indicates the blend of source pixels you want in the four tap filter kernal to provide to generate the first display pixel. The ov0setup.cpp program shows how to calculate the correct value.
OV0_PRESHIFT_P23_TO	30:28	0x2	Indicates how many source pixels you want to preshift into the horizontal U and V four tap filters to generate the first display pixel. Please refer to the documentation for sample code on how to program this field. Typically this value is two or three.

Please refer to P1_H_ACCUM_INIT. This register is used for the UV components in Planer modes.

OV0_P1_X_START_END - RW - 32 bits - [MMReg:0x494]			
Field Name	**Bits**	**Default**	**Description**
OV0_P1_X_END	11:0	0x0	End pixel in an octword aligned line for Y, R, G, and B
OV0_P1_X_START	19:16	0x0	Start pixel in an octword aligned line for Y, R, G, and B

Fields in the OV0_VID_BUF?_BASE_ADRS and OV0_VID_BUF_PITCH?_VALUE registers are used to determine the address of the start octword for each line. The P?_X_START_END registers indicate more accurately where the lines start and ends relative to that first octword. The P?_X_START field indicates exactly which pixel in the first octword to start on. The P?_X_END field indicates which pixel to end on relative to pixel zero in the first octword. The hardware uses the P?_X_END and the SURFACE_FORMAT fields to determine how many octwords to fetch.

This register allows pixel accurate windowing of the source surface which was not supported in earlier chips. It is now possible to zoom, pan (for DVD pan and scan), and clip the source.

Refer to ov0setup.cpp for sample code on how to program this register. You will notice the the CalcH.cpp routine returns values, P1StepSize and P23StepSize, that indicate the horizontal step when downscaling. These are used to insure that the start and end values are programmed correctly if pixel dropping occurs during horizontal downscaling.

P1 refers to Y, R, G, and B

OV0_P2_X_START_END - RW - 32 bits - [MMReg:0x498]			
Field Name	**Bits**	**Default**	**Description**
OV0_P2_X_END	10:0	0x0	End pixel in an octword aligned line for U.
OV0_P2_X_START	19:16	0x0	Start pixel in an octword aligned line for U

See the P1 description. This register (and its fields) will be renamed OV0_P23_X_START_END in the near future.

P2 refers to Cb (U)

OV0_P3_X_START_END - RW - 32 bits - [MMReg:0x49C]			
Field Name	**Bits**	**Default**	**Description**
OV0_P3_X_END	10:0	0x0	End pixel in an octword aligned line for V (program to the same value as OV0_P2_X_END if this register still exists.)
OV0_P3_X_START	19:16	0x0	Start pixel in an octword aligned line for V (program to the same value as OV0_P2_X_START if this register still exists.)

See the P1 description. Note that you can no longer program the P2(U) and P3(V) start and end points to different values. To support planer surfaces where the U and V planes to share a pitch, make sure that the pitch is a multiple of two octwords. This register is now redundant and will be removed at the earliest opportunity.

P3 refers to Cr (U)

OV0_FILTER_CNTL - RW - 32 bits - [MMReg:0x4A0]			
Field Name	**Bits**	**Default**	**Description**
OV0_HC_COEF_ON_HORZ_Y	0	0x0	Selects the hard coded upscaling filter coefficients for scaling the Y component horizontally. 0=PROGRAMABLE 1=HARD CODED
OV0_HC_COEF_ON_HORZ_UV	1	0x0	Selects the hard coded upscaling filter coefficients for scaling the UV component horizontally. 0=PROGRAMABLE 1=HARD CODED
OV0_HC_COEF_ON_VERT_Y	2	0x0	Selects the hard coded upscaling filter coefficients for scaling the Y component vertically. Applies only if the vertical filter engine is performing four tap filtering on the Y. 0=PROGRAMABLE 1=HARD CODED
OV0_HC_COEF_ON_VERT_UV	3	0x0	Selects the hard coded upscaling filter coefficients for scaling the UV component vertically. Applies only if the vertical filter engine is performing four tap filtering on the UV. 0=PROGRAMABLE 1=HARD CODED

There are five four tap filters. It is expensive to provide each filter with it's own set of programmable coefficients, so there are hard coded coefficients suitible for upscaling built into the hardware. For example, if you have to use the programable coefficients to specify a guassian curve for dowscaling Y horizontally, but you still want to upsample the UV using the sharper upscaling truncated sync function, then use the hard coded coefficients for the horizontal UV.

OV0_FOUR_TAP_COEF_0 - RW - 32 bits - [MMReg:0x4B0]			
Field Name	**Bits**	**Default**	**Description**
OV0_COEF_0TH_TAP	3:0	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.
OV0_COEF_1ST_TAP	14:8	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_2ND_TAP	22:16	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_3RD_TAP	27:24	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.

Coefficients used for phase 0/8. 0TH_TAP means the left most of top most pixel in a set of four will be multiplied by this coeefficient.

OV0_FOUR_TAP_COEF_1 - RW - 32 bits - [MMReg:0x4B4]			
Field Name	**Bits**	**Default**	**Description**
OV0_COEF_0TH_TAP	3:0	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.
OV0_COEF_1ST_TAP	14:8	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_2ND_TAP	22:16	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_3RD_TAP	27:24	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.

Coefficients used for phase 1/8 and 5/8 when swapped. 0TH_TAP means the left most of top most pixel in a set of four will be multiplied by this coeeficient when the coefficients aren't swapped.

OV0_FOUR_TAP_COEF_2 - RW - 32 bits - [MMReg:0x4B8]			
Field Name	**Bits**	**Default**	**Description**
OV0_COEF_0TH_TAP	3:0	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.
OV0_COEF_1ST_TAP	14:8	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_2ND_TAP	22:16	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_3RD_TAP	27:24	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.

Coefficients used for phase 2/8 and 6/8 when swapped. 0TH_TAP means the left most of top most pixel in a set of four will be multiplied by this coeeficient when the coefficients aren't swapped.

OV0_FOUR_TAP_COEF_3 - RW - 32 bits - [MMReg:0x4BC]			
Field Name	**Bits**	**Default**	**Description**
OV0_COEF_0TH_TAP	3:0	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.
OV0_COEF_1ST_TAP	14:8	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_2ND_TAP	22:16	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_3RD_TAP	27:24	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.

Coefficients used for phase 3/8 and 7/8 when swapped. 0TH_TAP means the left most of top most pixel in a set of four will be multiplied by this coeeficient when the coefficients aren't swapped.

OV0_FOUR_TAP_COEF_4 - RW - 32 bits - [MMReg:0x4C0]			
Field Name	**Bits**	**Default**	**Description**
OV0_COEF_0TH_TAP	3:0	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.
OV0_COEF_1ST_TAP	14:8	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.
OV0_COEF_2ND_TAP	22:16	0x0	This filter coefficient is a four bit signed value in the range -8 to +47.

OV0_COEF_3RD_TAP	27:24	0x0	This filter coefficient is a four bit signed value in the range of -8 to +7.

Coefficients used for phase 4/8. 0TH_TAP means the left most of top most pixel in a set of four will be multiplied by this coeefficient.

OV0_FLAG_CNTRL - RW - 32 bits - [MMReg:0x4DC]			
Field Name	**Bits**	**Default**	**Description**
OV0_HI_PRI_MCREQ	3:0	0x8	
OV0_HI_PRI_FORCE	8	0x0	0=NO FORCE 1=FORCE
OV0_LUMA_10BIT_EN	12	0x0	0=H filter produces 8 bits for luma 1=H filter produces 10 bits for luma
OV0_CHROMA_10BIT_EN	13	0x0	0=H filter produces 8 bits for chroma 1=H filter produces 10 bits for chroma

No description available for this register.

OV0_SLICE_CNTL - RW - 32 bits - [MMReg:0x4E0]			
Field Name	**Bits**	**Default**	**Description**
OV0_SLICE_LAST_LINE	6:0	0x0	
OV0_SLICEDONE_STAT (R)	30	0x0	0=Not done with slice. 1=Slice is done.
OV0_MPEG_EOF_TOGGLE	31	0x0	

No description available for this register.

OV0_VID_KEY_CLR_LOW - RW - 32 bits - [MMReg:0x4E4]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_KEY_Cb_BLUE_LOW	9:0	0x0	Overlay/subpicture keyer Cb or blue component lower limit.
OV0_VID_KEY_Y_GREEN_LOW	19:10	0x0	Overlay/subpicture keyer Y or green component lower limit.
OV0_VID_KEY_Cr_RED_LOW	29:20	0x0	Overlay/subpicture keyer Cr or red component lower limit.

Overlay/subpicture range keyer lower limits. The overlay/subpicture blend unit output where keying is done is 30bpp. This is the data before color conversion or gamma adjustment.

OV0_VID_KEY_CLR_HIGH - RW - 32 bits - [MMReg:0x4E8]			
Field Name	**Bits**	**Default**	**Description**
OV0_VID_KEY_Cb_BLUE_HIGH	9:0	0x0	Overlay/subpicture keyer Cb or blue component upper limit.
OV0_VID_KEY_Y_GREEN_HIGH	19:10	0x0	Overlay/subpicture keyer Y or green component upper limit.
OV0_VID_KEY_Cr_RED_HIGH	29:20	0x0	Overlay/subpicture keyer Cr or red component upper limit.

Overlay/subpicture range keyer upper limits. The overlay/subpicture blend unit output where keying is done is 30bpp. This is the data before color conversion or gamma adjustment.

OV0_GRPH_KEY_CLR_LOW - RW - 32 bits - [MMReg:0x4EC]			
Field Name	**Bits**	**Default**	**Description**
OV0_GRPH_KEY_BLUE_LOW	7:0	0x0	Graphics keyer blue component lower limit.
OV0_GRPH_KEY_GREEN_LOW	15:8	0x0	Graphics keyer green component lower limit.
OV0_GRPH_KEY_RED_LOW	23:16	0x0	Graphics keyer red component lower limit.
OV0_GRPH_KEY_ALPHA_LOW	31:24	0x0	Graphics keyer alpha component lower limit.

Graphics range keyer lower limits. Note in some modes expansion is needed to create an 8 bit red, green, blue or alpha from the 4, 5, or 6 bits stored in the frame buffer. This is controlled by DAC_CNTL.DAC_EXPAND_MODE. Graphics keying is always done on 8 bit post-expanded data. For pixel depths that do not have alpha, set the ALPHA_LOW to 0x00. To key on only the alpha channel in 15 or 32 bpp, set the color lower limits to 0x00.

OV0_GRPH_KEY_CLR_HIGH - RW - 32 bits - [MMReg:0x4F0]			
Field Name	**Bits**	**Default**	**Description**
OV0_GRPH_KEY_BLUE_HIGH	7:0	0x0	Graphics keyer blue component upper limit.
OV0_GRPH_KEY_GREEN_HIGH	15:8	0x0	Graphics keyer green component upper limit.
OV0_GRPH_KEY_RED_HIGH	23:16	0x0	Graphics keyer red component upper limit.
OV0_GRPH_KEY_ALPHA_HIGH	31:24	0x0	Graphics keyer alpha component upper limit.

Graphics range keyer upper limits. Note in some modes expansion is needed to create an 8 bit red, green, blue or alpha from the 4, 5, or 6 bits stored in the frame buffer. This is controlled by DAC_CNTL.DAC_EXPAND_MODE. Graphics keying is always done on 8 bit post-expanded data. For pixel depths that do not have alpha, set the ALPHA_HIGH to 0xFF. To key on only the alpha channel in 15 or 32 bpp, set the color high limits to 0xFF.

OV0_KEY_CNTL - RW - 32 bits - [MMReg:0x4F4]			
Field Name	**Bits**	**Default**	**Description**
OV0_VIDEO_KEY_FN	1:0	0x0	Selects overlay keyer result equation. Result used as stated in 0V0_CMP_MIX. 0=VID_KEY = FALSE = 0 1=VID_KEY = TRUE = 1 2=VID_KEY = (VID_Cr_BLUE in range) AND (VID_Cb_GREEN in range) AND (VID_Y_RED in range) 3=VID_KEY = not [(VID_Cr_BLUE in range) AND (VID_Cb_GREEN in range) AND (VID_Y_RED in range)]
OV0_GRAPHICS_KEY_FN	5:4	0x0	Selects graphics keyer result equation. Result used as stated in 0V0_CMP_MIX. 0=GRPH_KEY = FALSE = 0 1=GRPH_KEY = TRUE = 1 2=GPPH_KEY = (GRPH_BLUE in range) AND (GRPH_GREEN in range) AND (GRPH_RED in range) AND (GRPH_ALPHA in range) 3=GRPH_KEY = not [(GRPH_BLUE in range) AND (GRPH_GREEN in range) AND (GRPH_RED in range) AND (GRPH_ALPHA in range)]
OV0_CMP_MIX	8	0x0	Selects final mix of graphics and video keys. Result used as stated in DISP_MERGE_CNTL.DISP_ALPHA_MODE. 0=GRPH_VID_KEY = GRPH_KEY or VID_KEY 1=GRPH_VID_KEY = GRPH_KEY and VID_KEY

Graphics/overlay keyer control

OV0_TEST - RW - 32 bits - [MMReg:0x4F8]			
Field Name	**Bits**	**Default**	**Description**
OV0_MIN_OFFSET_EN	1	0x0	
OV0_SUBPIC_ONLY	3	0x0	Forces the subpicture data on. 0=NORMAL 1=PURE SUBPICTURE DATA
OV0_SWAP_UV	5	0x0	This field will swap the U and V components (and also the R and B components in RGB modes). 0=NO SWAP 1=SWAP
OV0_NOROUNDUP	6	0x0	0=ROUND UP HORZ & VERT FIR 1=DO NOT ROUND UP HORZ & VERT FIR
OV0_ADAPTIVE_DEINT_ADJ	13:12	0x0	
OV0_READ_BEHIND_SCAN_MARGIN	22:16	0x0	

Test registers for the scaler.

OV0_LIN_TRANS_A - RW - 32 bits - [MMReg:0xD20]			
Field Name	**Bits**	**Default**	**Description**
OV0_LIN_TRANS_Cb_R	15:4	0x0	Overlay linear transform unit coefficient for Cb of R. Format S3.8 (-8.00..+7.99)
OV0_LIN_TRANS_Y_R	31:20	0x12a	Overlay linear transform unit coefficient for Y of R. Format S3.8 (-8.00..+7.99)

Overlay linear transform unit coefficient group A

OV0_LIN_TRANS_B - RW - 32 bits - [MMReg:0xD24]			
Field Name	**Bits**	**Default**	**Description**
OV0_LIN_TRANS_OFF_R	12:0	0x190e	Overlay linear transform unit coefficient offset of R. Format S11.1 (-2048.0..+2047.5)
OV0_LIN_TRANS_Cr_R	31:20	0x199	Overlay linear transform unit coefficient for Cr of R. Format S3.8 (-8.00..+7.99)

Overlay linear transform unit coefficient group B

OV0_LIN_TRANS_C - RW - 32 bits - [MMReg:0xD28]			
Field Name	**Bits**	**Default**	**Description**
OV0_LIN_TRANS_Cb_G	15:4	0xf9c	Overlay linear transform unit coefficient for Cb of G. Format S3.8 (-8.00..+7.99)
OV0_LIN_TRANS_Y_G	31:20	0x12a	Overlay linear transform unit coefficient for Y of G. Format S3.8 (-8.00..+7.99)

Overlay linear transform unit coefficient group C

OV0_LIN_TRANS_D - RW - 32 bits - [MMReg:0xD2C]			
Field Name	**Bits**	**Default**	**Description**
OV0_LIN_TRANS_OFF_G	12:0	0x442	Overlay linear transform unit coefficient offset of G. Format S11.1 (-2048.0..+2047.5)
OV0_LIN_TRANS_Cr_G	31:20	0xf30	Overlay linear transform unit coefficient for Cr of G. Format S3.8 (-8.00..+7.99)

Overlay linear transform unit coefficient group D

OV0_LIN_TRANS_E - RW - 32 bits - [MMReg:0xD30]			
Field Name	**Bits**	**Default**	**Description**
OV0_LIN_TRANS_Cb_B	15:4	0x204	Overlay linear transform unit coefficient for Cb of B. Format S3.8 (-8.00..+7.99)
OV0_LIN_TRANS_Y_B	31:20	0x12a	Overlay linear transform unit coefficient for Y of B. Format S3.8 (-8.00..+7.99)

Overlay linear transform unit coefficient group E

OV0_LIN_TRANS_F - RW - 32 bits - [MMReg:0xD34]			
Field Name	**Bits**	**Default**	**Description**
OV0_LIN_TRANS_OFF_B	12:0	0x175f	Overlay linear transform unit coefficient offset of B. Format S11.1 (-2048.0..+2047.5)
OV0_LIN_TRANS_Cr_B	31:20	0x0	Overlay linear transform unit coefficient for Cr of B. Format S3.8 (-8.00..+7.99)

Overlay linear transform unit coefficient group F

OV0_GAMMA_0_F - RW - 32 bits - [MMReg:0xD40]			
Field Name	**Bits**	**Default**	**Description**
OV0_GAMMA_0_F_OFFSET	8:0	0x0	Overlay gamma correction non-linear offset for input 0-F. Format 8.1 (0.0..+255.5)
OV0_GAMMA_0_F_SLOPE	26:16	0x100	Overlay gamma correction non-linear slope for input 0-F. Format 3.8 (0.00..+7.99)

Overlay gamma correction non-linear offset and slope for input 0-F

OV0_GAMMA_10_1F - RW - 32 bits - [MMReg:0xD44]			
Field Name	**Bits**	**Default**	**Description**
OV0_GAMMA_10_1F_OFFSET	8:0	0x20	Overlay gamma correction non-linear offset for input 10-1F. Format 8.1 (0.0..+255.5)
OV0_GAMMA_10_1F_SLOPE	26:16	0x100	Overlay gamma correction non-linear slope for input 10-1F. Format 3.8 (0.00..+7.99)

Overlay gamma correction non-linear offset and slope for input 10-1F

OV0_GAMMA_20_3F - RW - 32 bits - [MMReg:0xD48]			
Field Name	**Bits**	**Default**	**Description**
OV0_GAMMA_20_3F_OFFSET	9:0	0x40	Overlay gamma correction non-linear offset for input 20-3F. Format 9.1 (0.0..+511.5)
OV0_GAMMA_20_3F_SLOPE	25:16	0x100	Overlay gamma correction non-linear slope for input 20-3F. Format 2.8 (0.00..+3.99)

Overlay gamma correction non-linear offset and slope for input 20-3F

OV0_GAMMA_40_7F - RW - 32 bits - [MMReg:0xD4C]			
Field Name	**Bits**	**Default**	**Description**
OV0_GAMMA_40_7F_OFFSET	9:0	0x80	Overlay gamma correction non-linear offset for input 40-7F. Format 9.1 (0.0..+511.5)
OV0_GAMMA_40_7F_SLOPE	24:16	0x100	Overlay gamma correction non-linear slope for input 40-7F. Format 1.8 (0.00..+1.99)

Overlay gamma correction non-linear offset and slope for input 40-7F

OV0_GAMMA_380_3BF - RW - 32 bits - [MMReg:0xD50]			
Field Name	**Bits**	**Default**	**Description**
OV0_GAMMA_380_3BF_OFFSET	8:0	0x100	Overlay gamma correction non-linear offset for input 380-3BF. Format 8.1 + 0x300 integer added by HW (+768.0..+1023.5)
OV0_GAMMA_380_3BF_SLOPE	24:16	0x100	Overlay gamma correction non-linear slope for input 380-3BF. Format 1.8 (0.00..+1.99)

Overlay gamma correction non-linear offset and slope for input 380-3BF

OV0_GAMMA_3C0_3FF - RW - 32 bits - [MMReg:0xD54]			
Field Name	**Bits**	**Default**	**Description**
OV0_GAMMA_3C0_3FF_OFFSET	8:0	0x100	Overlay gamma correction non-linear offset for input 3C0-3FF. Format 8.1 + 0x300 integer added by HW (+768.0..+1023.5)
OV0_GAMMA_3C0_3FF_SLOPE	24:16	0x100	Overlay gamma correction non-linear slope for input 3C0-3FF. Format 1.8 (0.00..+1.99)

Overlay gamma correction non-linear offset and slope for input 3C0-3FF

OV1_Y_X_START - RW - 32 bits - [MMReg:0x600]			
Field Name	**Bits**	**Default**	**Description**
OV1_X_START	12:0	0x0	Overlay X ending coordinate relative to secondary ACTIVE screen (0,0)
OV1_Y_START	28:16	0x0	Overlay Y ending coordinate relative to secondary ACTIVE screen (0,0)

X/Y coords relative to active secondary display in pixels

OV1_Y_X_END - RW - 32 bits - [MMReg:0x604]			
Field Name	**Bits**	**Default**	**Description**
OV1_X_END	12:0	0x0	Overlay X ending coordinate relative to secondary ACTIVE screen (0,0)
OV1_Y_END	28:16	0x0	Overlay X ending coordinate relative to secondary ACTIVE screen (0,0)

X/Y ending coordinates (inclusive) relative to secondary active display in pixels

OV1_PIPELINE_CNTL - RW - 32 bits - [MMReg:0x608]			
Field Name	**Bits**	**Default**	**Description**
OV1_DISP_PIPE_DELAY	3:0	0x8	Secondary display debug field used to adjust overlay window position. Leave at default unless hardware problem requires adjustment as instructed.

Secondary display debug adjustments for overlay window control

2.25 cursor Registers

CUR_OFFSET - RW - 32 bits - [MMReg:0x260]			
Field Name	**Bits**	**Default**	**Description**
CUR_OFFSET	26:0	0x0	Hardware cursor address offset. For monochrome cursors (CRTC_CUR_MODE = 00), this vlaue must be a 16 byte aligned address. For color cursor modes it must be a 256 byte aligned address. The CUR_OFFSET is realative to the DISP_BASE_ADDRESS, meaning the active display pointed to by CRTC_OFFSET and the cursor must be in the same region of memory. This value is adjusted to move the cursor off the top edge of the display. See the CUR_VERT_OFF description. NOTE: Bits 0:3 of this field are hardwired to ZERO.
CUR_LOCK	31	0x0	Locks the HWC_EN, CUR_OFFSET, CUR_HORZ_VERT_POSN and CUR_HORZ_VERT_OFF registers to allow tear free atomic updating of the cursor shape and/or position. Moving the cursor around on the top and/or left edges, or changing the shape, requires multiple register writes. If these were done without setting CUR_LOCK, then flicker could occur. 0=Unlocked 1=Locked

Location of the hardware cursor image.

CUR_HORZ_VERT_POSN - RW - 32 bits - [MMReg:0x264]			
Field Name	**Bits**	**Default**	**Description**
CUR_VERT_POSN	11:0	0x0	Cursor vertical position. To move the cursor off the top edge set CUR_VERT_POSN=0 and see the CUR_VERT_OFF description. When CRTC_GEN_CNTL.CRTC_DBL_SCAN_EN=1 the logical position from the OS must be converter to a physical position to program here by multiplying by 2. i.e. no odd values should be used in double scan mode.
CUR_HORZ_POSN	29:16	0x0	Cursor horizontal position. To move the cursor off the left edge set CUR_HORZ_POSN=0 and see the CUR_HORZ_OFF description.

CUR_LOCK *(mirror of CUR_OFFSET:CUR_LOCK)*	31	0x0	Locks the HWC_EN, CUR_OFFSET, CUR_HORZ_VERT_POSN and CUR_HORZ_VERT_OFF registers to allow tear free atomic updating of the cursor shape and/or position. Moving the cursor around on the top and/or left edges, or changing the shape, requires multiple register writes. If these were done without setting CUR_LOCK, then flicker could occur. 0=Unlocked 1=Locked

Sets the screen position of the top left pixel of the visible part of the hardware cursor.

CUR_HORZ_VERT_OFF - RW - 32 bits - [MMReg:0x268]			
Field Name	**Bits**	**Default**	**Description**
CUR_VERT_OFF	5:0	0x0	Cursor vertical offset. Height of cursor is (64-CUR_VERT_OFF). To move the cursor off the top of the display, set CUR_VERT_POSN to 0, add 16*(number of lines to move off the top) to CUR_OFFSET, and increase CUR_VERT_OFF by the same number of lines. When is double scan display (CRTC_DBL_SCAN_EN=1) the height of the cursor is limited to 32 logical lines, but 64 physical lines. i.e. the image in memory can be up to 32 lines high, but will cover 64 physical lines of the display. Also in double scan mode the height of the cursor must be even in physical lines. i.e. this field must be even.
CUR_HORZ_OFF	21:16	0x0	Cursor horizontal offset. Width of the cursor is always 64 pixels. CUR_HORZ_OFF controls how far into the cursor map from the left is 'pixel 0'. The horizontal position on the display of 'pixel 0' is set by CUR_HORZ_POSN. Therefore to move the cursor off the left edge of the display, set the CUR_HORZ_POSN to zero, and increase the CUR_HORZ_OFF by the number of pixels off the left edge.
CUR_LOCK *(mirror of CUR_OFFSET:CUR_LOCK)*	31	0x0	Locks the HWC_EN, CUR_OFFSET, CUR_HORZ_VERT_POSN and CUR_HORZ_VERT_OFF registers to allow tear free atomic updating of the cursor shape and/or position. Moving 0=Unlocked 1=Locked

Controls the size of the hardware cursor mask in memory, and used to move the cursor off the top and/or left edges of the display.

CUR_CLR0 - RW - 32 bits - [MMReg:0x26C]			
Field Name	**Bits**	**Default**	**Description**
CUR_CLR0_B	7:0	0x0	Blue component of cursor color 0.
CUR_CLR0_G	15:8	0x0	Green component of cursor color 0.
CUR_CLR0_R	23:16	0x0	Red component of cursor color 0.

Location of the hardware cursor image.

CUR_CLR1 - RW - 32 bits - [MMReg:0x270]			
Field Name	**Bits**	**Default**	**Description**
CUR_CLR1_B	7:0	0x0	Blue component of cursor color 1.
CUR_CLR1_G	15:8	0x0	Green component of cursor color 1.
CUR_CLR1_R	23:16	0x0	Red component of cursor color 1.

Hardware cursor color 1. Always 24bpp, independent of graphics mode.
For monochrome cursor only.

CUR2_OFFSET - RW - 32 bits - [MMReg:0x360]			
Field Name	**Bits**	**Default**	**Description**
CUR2_OFFSET	26:0	0x0	Hardware cursor address offset. For monochrome cursors (CRTC2_CUR_MODE = 00), this vlaue must be a 16 byte aligned address. For color cursor modes it must be a 256 byte aligned address. The CUR_OFFSET is realative to the DISP_BASE_ADDRESS, meaning the active display pointed to by CRTC2_OFFSET and the cursor must be in the same region of memory. This value is adjusted to move the cursor off the top edge of the display. See the CUR_VERT_OFF description. NOTE: Bits 0:3 of this field are hardwired to ZERO.

CUR2_LOCK	31	0x0	Locks the HWC_EN, CUR_OFFSET, CUR_HORZ_VERT_POSN and CUR_HORZ_VERT_OFF registers to allow tear free atomic updating of the cursor shape and/or position. Moving the cursor around on the top and/or left edges, or changing the shape, requires multiple register writes. If these were done without setting CUR_LOCK, then flicker could occur. 0=Unlocked 1=Locked

Location of the secondary hardware cursor image.

CUR2_HORZ_VERT_POSN - RW - 32 bits - [MMReg:0x364]			
Field Name	**Bits**	**Default**	**Description**
CUR2_VERT_POSN	11:0	0x0	Cursor vertical position. To move the cursor off the top edge set CUR_VERT_POSN=0 and see the CUR_VERT_OFF description. When CRTC2_GEN_CNTL.CRTC2_DBL_SCAN_EN=1 the logical position from the OS must be converter to a physical position to program here by multiplying by 2. i.e. no odd values should be used in double scan mode.
CUR2_HORZ_POSN	29:16	0x0	Cursor horizontal position. To move the cursor off the left edge set CUR_HORZ_POSN=0 and see the CUR_HORZ_OFF description.
CUR2_LOCK *(mirror of CUR2_OFFSET:CUR2_LOCK)*	31	0x0	Locks the HWC_EN, CUR_OFFSET, CUR_HORZ_VERT_POSN and CUR_HORZ_VERT_OFF registers to allow tear free atomic updating of the cursor shape and/or position. Moving the cursor around on the top and/or left edges, or changing the shape, requires multiple register writes. If these were done without setting CUR_LOCK, then flicker could occur. 0=Unlocked 1=Locked

Sets the screen position of the top left pixel of the visible part of the secondary hardware cursor.

CUR2_HORZ_VERT_OFF - RW - 32 bits - [MMReg:0x368]			
Field Name	**Bits**	**Default**	**Description**
CUR2_VERT_OFF	5:0	0x0	Cursor vertical offset. Height of cursor is (64-CUR_VERT_OFF). To move the cursor off the top of the display, set CUR_VERT_POSN to 0, add 16*(number of lines to move off the top) to CUR_OFFSET, and increase CUR_VERT_OFF by the same number of lines. When is double scan display (CRTC2_DBL_SCAN_EN=1) the height of the cursor is limited to 32 logical lines, but 64 physical lines. i.e. the image in memory can be up to 32 lines high, but will cover 64 physical lines of the display. Also in double scan mode the height of the cursor must be even in physical lines. i.e. this field must be even.
CUR2_HORZ_OFF	21:16	0x0	Cursor horizontal offset. Width of the cursor is always 64 pixels. CUR_HORZ_OFF controls how far into the cursor map from the left is 'pixel 0'. The horizontal position on the display of 'pixel 0' is set by CUR_HORZ_POSN. Therefore to move the cursor off the left edge of the display, set the CUR_HORZ_POSN to zero, and increase the CUR_HORZ_OFF by the number of pixels off the left edge.
CUR2_LOCK *(mirror of CUR2_OFFSET:CUR2_LOCK)*	31	0x0	Locks the HWC_EN, CUR_OFFSET, CUR_HORZ_VERT_POSN and CUR_HORZ_VERT_OFF registers to allow tear free atomic updating of the cursor shape and/or position. Moving 0=Unlocked 1=Locked

Controls the size of the secondary hardware cursor mask in memory, and used to move the cursor off the top and/or left edges of the display.

CUR2_CLR0 - RW - 32 bits - [MMReg:0x36C]			
Field Name	**Bits**	**Default**	**Description**
CUR2_CLR0_B	7:0	0x0	Blue component of cursor color 0.
CUR2_CLR0_G	15:8	0x0	Green component of cursor color 0.
CUR2_CLR0_R	23:16	0x0	Red component of cursor color 0.

Location of the secondary hardware cursor image.

CUR2_CLR1 - RW - 32 bits - [MMReg:0x370]			
Field Name	**Bits**	**Default**	**Description**
CUR2_CLR1_B	7:0	0x0	Blue component of cursor color 1.
CUR2_CLR1_G	15:8	0x0	Green component of cursor color 1.
CUR2_CLR1_R	23:16	0x0	Red component of cursor color 1.

Hardware cursor color 1. Always 24bpp, independent of graphics mode.
For monochrome cursor only.

2.26 overscan Registers

OVR2_CLR - RW - 32 bits - [MMReg:0x330]			
Field Name	**Bits**	**Default**	**Description**
OVR2_CLR_B	7:0	0x0	Blue component overscan colour.
OVR2_CLR_G	15:8	0x0	Green component overscan colour.
OVR2_CLR_R	23:16	0x0	Red component overscan colour.

Secondary display Overscan color. Always 24 bit, independent of pixel depth.

OVR2_WID_LEFT_RIGHT - RW - 32 bits - [MMReg:0x334]			
Field Name	**Bits**	**Default**	**Description**
OVR2_WID_RIGHT	6:0	0x0	Right overscan width (in pixels * 8).
OVR2_WID_LEFT	22:16	0x0	Left overscan width (in pixels * 8).

Secondary display Overscan color. Always 24 bit, independent of pixel depth.

OVR2_WID_TOP_BOTTOM - RW - 32 bits - [MMReg:0x338]			
Field Name	**Bits**	**Default**	**Description**
OVR2_WID_BOTTOM	9:0	0x0	Bottom overscan width (in scan lines).
OVR2_WID_TOP	25:16	0x0	Top overscan width (in scan lines).

Secondary display Overscan border top/bottom width control.

OVR_CLR - RW - 32 bits - [MMReg:0x230]			
Field Name	**Bits**	**Default**	**Description**
OVR_CLR_B	7:0	0x0	Blue component overscan colour.
OVR_CLR_G	15:8	0x0	Green component overscan colour.
OVR_CLR_R	23:16	0x0	Red component overscan colour.

Overscan color. Always 24 bit, independent of pixel depth.

OVR_WID_LEFT_RIGHT - RW - 32 bits - [MMReg:0x234]			
Field Name	**Bits**	**Default**	**Description**
OVR_WID_RIGHT	6:0	0x0	Right overscan width (in pixels * 8).
OVR_WID_LEFT	22:16	0x0	Left overscan width (in pixels * 8).

Overscan color. Always 24 bit, independent of pixel depth.

OVR_WID_TOP_BOTTOM - RW - 32 bits - [MMReg:0x238]			
Field Name	**Bits**	**Default**	**Description**
OVR_WID_BOTTOM	9:0	0x0	Bottom overscan width (in scan lines).
OVR_WID_TOP	25:16	0x0	Top overscan width (in scan lines).

Overscan border top/bottom width control.

2.27 hwicon Registers

ICON_OFFSET - RW - 32 bits - [MMReg:0x2B0]			
Field Name	**Bits**	**Default**	**Description**
ICON_OFFSET	26:0	0x0	Primary display hardware icon origin offset in memory. Must be 8 byte aligned for 4bpp, 8bpp and 24bpp modes. Must be 16 byte aligned for all 16bpp modes. Must be 32 byte aligned for 32bpp mode. When CRTC_STEREO_OFFSET_EN = 1, this is the left eye image. This is added to DISP_BASE_ADDR to get the full 32 bit address of the graphics surface. NOTE: Bits 0:4 of this field are hardwired to ZERO.
BLANK_SCREEN1	27	0x0	Blank screen except for ICON on primary display 0=Normal 1=Blank Screen Except for Icon on primary display
DONT_DS_ICON	28	0x0	Double scan primary hardware icon 0=Expand icon in double scan mode 1=Do not expand icon in double scan mode
ICON_LOCK	31	0x0	0=Unlocked 1=Locked

Location of primary display hardware icon image

ICON_HORZ_VERT_POSN - RW - 32 bits - [MMReg:0x2B4]			
Field Name	**Bits**	**Default**	**Description**
ICON_VERT_POSN	11:0	0x0	Icon vertical position
ICON_HORZ_POSN	27:16	0x0	Icon horizontal position
ICON_LOCK *(mirror of ICON_OFFSET:ICON_LOCK)*	31	0x0	0=Unlocked 1=Locked

Primary display hardware icon horizontal and vertical positions

ICON_HORZ_VERT_OFF - RW - 32 bits - [MMReg:0x2B8]			
Field Name	**Bits**	**Default**	**Description**
ICON_VERT_OFF	6:0	0x0	Icon vertical offset
ICON_HORZ_OFF	22:16	0x0	Icon horizontal offset
ICON_LOCK *(mirror of ICON_OFFSET:ICON_LOCK)*	31	0x0	0=Unlocked 1=Locked

Primary display hardware icon horizontal and vertical offsets

ICON_CLR0 - RW - 32 bits - [MMReg:0x2BC]			
Field Name	**Bits**	**Default**	**Description**
ICON_CLR0_B	7:0	0x0	Primary hardware icon blue colour 0
ICON_CLR0_G	15:8	0x0	Primary hardware icon green colour 0
ICON_CLR0_R	23:16	0x0	Primary hardware icon red colour 0

Primary hardware icon colour 0 register

ICON_CLR1 - RW - 32 bits - [MMReg:0x2C0]			
Field Name	**Bits**	**Default**	**Description**
ICON_CLR1_B	7:0	0x0	Primary hardware icon blue colour 1
ICON_CLR1_G	15:8	0x0	Primary hardware icon green colour 1
ICON_CLR1_R	23:16	0x0	Primary hardware icon red colour 1

Primary hardware icon colour 1 register

ICON2_OFFSET - RW - 32 bits - [MMReg:0x3B0]			
Field Name	**Bits**	**Default**	**Description**
ICON2_OFFSET	26:0	0x0	Secondary display hardware icon origin offset in memory. Must be 8 byte aligned for 4bpp, 8bpp and 24bpp modes. Must be 16 byte aligned for all 16bpp modes. Must be 32 byte aligned for 32bpp mode. This is added to DISP_BASE_ADDR to get the full 32 bit address of the graphics surface. NOTE: Bits 0:4 of this field are hardwired to ZERO.
BLANK_SCREEN2	27	0x0	Blank screen except for ICON on seconary display 0=Normal 1=Blank Screen Except for Icon on secondary display
DONT_DS_ICON2	28	0x0	Double scan seconary hardware icon 0=Expand icon in double scan mode 1=Do not expand icon in double scan mode
ICON2_LOCK	31	0x0	0=Unlocked 1=Locked

Location of secondary display hardware icon image

ICON2_HORZ_VERT_POSN - RW - 32 bits - [MMReg:0x3B4]			
Field Name	**Bits**	**Default**	**Description**
ICON2_VERT_POSN	11:0	0x0	Icon vertical position
ICON2_HORZ_POSN	27:16	0x0	Icon horizontal position
ICON2_LOCK *(mirror of ICON2_OFFSET:ICON2_LOCK)*	31	0x0	0=Unlocked 1=Locked

Secondary display hardware icon horizontal and vertical positions

ICON2_HORZ_VERT_OFF - RW - 32 bits - [MMReg:0x3B8]			
Field Name	**Bits**	**Default**	**Description**
ICON2_VERT_OFF	6:0	0x0	Icon vertical offset
ICON2_HORZ_OFF	22:16	0x0	Icon horizontal offset
ICON2_LOCK *(mirror of ICON2_OFFSET:ICON2_LOCK)*	31	0x0	0=Unlocked 1=Locked

Secondary display hardware icon horizontal and vertical offsets

ICON2_CLR0 - RW - 32 bits - [MMReg:0x3BC]			
Field Name	**Bits**	**Default**	**Description**
ICON2_CLR0_B	7:0	0x0	Secondary hardware icon blue colour 0
ICON2_CLR0_G	15:8	0x0	Secondary hardware icon green colour 0
ICON2_CLR0_R	23:16	0x0	Secondary hardware icon red colour 0

Secondary hardware icon colour 0 register

ICON2_CLR1 - RW - 32 bits - [MMReg:0x3C0]			
Field Name	**Bits**	**Default**	**Description**
ICON2_CLR1_B	7:0	0x0	Secondary hardware icon blue colour 1
ICON2_CLR1_G	15:8	0x0	Secondary hardware icon green colour 1
ICON2_CLR1_R	23:16	0x0	Secondary hardware icon red colour 1

Secondary hardware icon colour 1 register

2.28 subpic Registers

SUBPIC_CNTL - RW - 32 bits - [MMReg:0x540]			
Field Name	**Bits**	**Default**	**Description**
SUBPIC_ON	0	0x0	Turn on subpicture. This bit indicates the on/off status of the subpicture.
BTN_HLI_ON	1	0x0	Turn on Button Highlight. This bit indicates the on/off state of the Button Highlight feature
SP_HORZ_MODE	4	0x0	Subpicture Horizontal Mode 0=blend(default) 1=replicate
SP_VERT_MODE	5	0x0	Subpicture Vertical Mode 0=blend(default) 1=replicate
SP_ODD_FIELD	8	0x0	Current subpicture field 0=even 1=odd
SP_BUF_SELECT	9	0x0	Source buffer select 0=Buf0, 1=Buf1
SP_NO_R_EDGE_BLEND	10	0x0	Turn off blending on the right edge of the subpicture data 0=blending enabled(default) 1=blending disabled

No description available for this register.

SUBPIC_DEFCOLCON - RW - 32 bits - [MMReg:0x544]			
Field Name	**Bits**	**Default**	**Description**
BKGD_PIX_CON	3:0	0x0	Default Background Pixel Contrast
PATT_PIX_CON	7:4	0x0	Default Pattern Pixel Contrast
EMPH_PIX1_CON	11:8	0x0	Default Emphasis Pixel-1 Contrast
EMPH_PIX2_CON	15:12	0x0	Default Emphasis Pixel-2 Contrast
BKGD_PIX_CLR	19:16	0x0	Default Background Pixel Color
PATT_PIX_CLR	23:20	0x0	Default Pattern Pixel Color
EMPH_PIX1_CLR	27:24	0x0	Default Emphasis Pixel-1 Color
EMPH_PIX2_CLR	31:28	0x0	Default Emphasis Pixel-2 Color

No description available for this register.

SUBPIC_Y_X_START - RW - 32 bits - [MMReg:0x54C]			
Field Name	**Bits**	**Default**	**Description**
SP_START_X	9:0	0x0	Subpicture Start X-Coordinate
SP_START_Y	25:16	0x0	Subpicture Start Y-Coordinate

No description available for this register.

SUBPIC_Y_X_END - RW - 32 bits - [MMReg:0x550]			
Field Name	**Bits**	**Default**	**Description**
SP_END_X	9:0	0x0	Subpicture End X-Coordinate
SP_END_Y	25:16	0x0	Subpicture End Y-Coordinate

No description available for this register.

SUBPIC_V_INC - RW - 32 bits - [MMReg:0x554]			
Field Name	**Bits**	**Default**	**Description**
SP_V_INC_FRAC	15:4	0x0	Subpicture Vertical Increment Value (Fraction)
SP_V_INC_INT	19:16	0x0	Subpicture Vertical Increment Value (Integer)

No description available for this register.

SUBPIC_H_INC - RW - 32 bits - [MMReg:0x558]			
Field Name	**Bits**	**Default**	**Description**
SP_H_INC_FRAC	15:4	0x0	Horizontal Accumulator Increment (Fraction)
SP_H_INC_INT	19:16	0x0	Horizontal Accumulator Increment (Integer)

No description available for this register.

SUBPIC_BUF0_OFFSET - RW - 32 bits - [MMReg:0x55C]			
Field Name	**Bits**	**Default**	**Description**
SUBPIC_OFFSET0	31:0	0x0	Subpicture Buffer 0 Offset (4-Octword) NOTE: Bits 0:5 of this field are hardwired to ZERO.

No description available for this register.

SUBPIC_BUF1_OFFSET - RW - 32 bits - [MMReg:0x560]			
Field Name	**Bits**	**Default**	**Description**
SUBPIC_OFFSET1	31:0	0x0	Subpicture Buffer 1 Offset (4-Octword) NOTE: Bits 0:5 of this field are hardwired to ZERO.

No description available for this register.

SUBPIC_LC0_OFFSET - RW - 32 bits - [MMReg:0x564]			
Field Name	**Bits**	**Default**	**Description**
SUBPIC_LC0_OFFSET	31:0	0x0	Subpicture Line Change Offset 0 (4 Octword address) NOTE: Bits 0:5 of this field are hardwired to ZERO.

No description available for this register.

SUBPIC_LC1_OFFSET - RW - 32 bits - [MMReg:0x568]			
Field Name	**Bits**	**Default**	**Description**
SUBPIC_LC1_OFFSET	31:0	0x0	Subpicture Line Change Offset 1 (4 Octword address) NOTE: Bits 0:5 of this field are hardwired to ZERO.

No description available for this register.

SUBPIC_PITCH - RW - 32 bits - [MMReg:0x56C]			
Field Name	**Bits**	**Default**	**Description**
SUBPIC_BUF_PITCH	11:0	0x0	Subpicture Buffer Pitch (4 Octword Address) NOTE: Bits 0:5 of this field are hardwired to ZERO.
SUBPIC_LC_PITCH	27:16	0x0	Subpicture Line Control Pitch (4 Octword Address) NOTE: Bits 0:5 of this field are hardwired to ZERO.

No description available for this register.

SUBPIC_BTN_HLI_COLCON - RW - 32 bits - [MMReg:0x570]			
Field Name	**Bits**	**Default**	**Description**
BTN_HLI_BKGD_PIX_CON	3:0	0x0	Button Highlight Background Pixel Contrast
BTN_HLI_PATT_PIX_CON	7:4	0x0	Button Highlight Pattern Pixel Contrast
BTN_HLI_EMPH_PIX1_CON	11:8	0x0	Button Highlight Emphasis Pixel-1 Contrast
BTN_HLI_EMPH_PIX2_CON	15:12	0x0	Button Highlight Emphasis Pixel-2 Contrast
BTN_HLI_BKGD_PIX_CLR	19:16	0x0	Button Highlight Background Pixel Color
BTN_HLI_PATT_PIX_CLR	23:20	0x0	Button Highlight Pattern Pixel Color
BTN_HLI_EMPH_PIX1_CLR	27:24	0x0	Button Highlight Emphasis Pixel-1 Color
BTN_HLI_EMPH_PIX2_CLR	31:28	0x0	Button Highlight Emphasis Pixel-2 Color

No description available for this register.

SUBPIC_BTN_HLI_Y_X_START - RW - 32 bits - [MMReg:0x574]			
Field Name	**Bits**	**Default**	**Description**
BTN_HLI_START_X	9:0	0x0	Button Highlight Start X-coordinate
BTN_HLI_START_Y	25:16	0x0	Button Highlight Start Y-coordinate

No description available for this register.

SUBPIC_BTN_HLI_Y_X_END - RW - 32 bits - [MMReg:0x578]			
Field Name	**Bits**	**Default**	**Description**
BTN_HLI_END_X	9:0	0x0	Button Highlight End X-coordinate
BTN_HLI_END_Y	25:16	0x0	Button Highlight End Y-Coordinate

No description available for this register.

SUBPIC_PALETTE_INDEX - RW - 32 bits - [MMReg:0x57C]			
Field Name	**Bits**	**Default**	**Description**
SP_PAL_ADDR	3:0	0x0	Subpicture Palette Address

No description available for this register.

SUBPIC_PALETTE_DATA - RW - 32 bits - [MMReg:0x580]			
Field Name	**Bits**	**Default**	**Description**
SP_DATA	23:0	0x0	Subpicture Palette Data

No description available for this register.

SUBPIC_0_PAL - RW - 32 bits - SUBPICIND:0x0			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y data

No description available for this register.

SUBPIC_1_PAL - RW - 32 bits - SUBPICIND:0x1			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_2_PAL - RW - 32 bits - SUBPICIND:0x2			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_3_PAL - RW - 32 bits - SUBPICIND:0x3			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data

SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_4_PAL - RW - 32 bits - SUBPICIND:0x4			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_5_PAL - RW - 32 bits - SUBPICIND:0x5			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_6_PAL - RW - 32 bits - SUBPICIND:0x6			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_7_PAL - RW - 32 bits - SUBPICIND:0x7			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data

SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_8_PAL - RW - 32 bits - SUBPICIND:0x8			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_9_PAL - RW - 32 bits - SUBPICIND:0x9			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_A_PAL - RW - 32 bits - SUBPICIND:0xA			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_B_PAL - RW - 32 bits - SUBPICIND:0xB			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data

SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_C_PAL - RW - 32 bits - SUBPICIND:0xC			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_D_PAL - RW - 32 bits - SUBPICIND:0xD			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_E_PAL - RW - 32 bits - SUBPICIND:0xE			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_F_PAL - RW - 32 bits - SUBPICIND:0xF			
Field Name	**Bits**	**Default**	**Description**
SP_CB	7:0	0x0	Palette Cb Data
SP_CR	15:8	0x0	Palette Cr Data
SP_Y	23:16	0x0	Palette Y Data

No description available for this register.

SUBPIC_H_ACCUM_INIT - RW - 32 bits - [MMReg:0x584]			
Field Name	**Bits**	**Default**	**Description**
SP_H_ACC_INIT_FRAC	15:4	0x0	Horizontal Accumulator Initial Value (Fraction)
SP_H_ACC_INIT_INT	26:16	0x0	Horizontal Accumulator Initial Value (Integer)

No description available for this register.

SUBPIC_V_ACCUM_INIT - RW - 32 bits - [MMReg:0x588]			
Field Name	**Bits**	**Default**	**Description**
SP_V_ACC_INIT_FRAC	15:4	0x0	Subpicture Vertical Accumulator Init Value (Fraction)
SP_V_ACC_INIT_INT	26:16	0x0	Subpicture Vertical Accumulator Init Value (Integer)

No description available for this register.

2.29 clkcntl Registers

CLOCK_CNTL_INDEX - RW - 32 bits - [IOReg,MMReg:0x8]			
Field Name	**Bits**	**Default**	**Description**
PLL_ADDR (DISPLAY) (CG) (CGM)	5:0	0x0	Indirect CG and PLL register index
PLL_WR_EN (DISPLAY) (CG) (CGM)	7	0x0	Indirect CG and PLL register write enable 0=Disable writes to CLOCK_CNTL_DATA 1=Enable writing to CLOCK_CNTL_DATA
PPLL_DIV_SEL (DISPLAY) (CG) (CGM)	9:8	0x0	Pixel clock PLL feedback division selection for non-VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=1. See GENMO_WT.VGA_CKSEL for pixel clock selection when VGA mode active. 0=PPLL_DIV0 1=PPLL_DIV1 2=PPLL_DIV2 3=PPLL_DIV3

CG and PLL indirect register control

CLOCK_CNTL_DATA - RW - 32 bits - [IOReg,MMReg:0xC]			
Field Name	**Bits**	**Default**	**Description**
PLL_DATA (DISPLAY) (CG) (CGM)	31:0	0x0	Data for write to CG and PLL indirect registers

Data for write to CG and PLL indirect registers

PPLL_DIV_0 - RW - 32 bits - CLKIND:0x4			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB0_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST0_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 0

PPLL_DIV_1 - RW - 32 bits - CLKIND:0x5			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB1_DIV (CG)	10:0	0x1f2	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.

PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST1_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 1

PPLL_DIV_2 - RW - 32 bits - CLKIND:0x6			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB2_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST2_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 2

PPLL_DIV_3 - RW - 32 bits - CLKIND:0x7			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB3_DIV (CG)	10:0	0x1bb	Feedback divider for pixel clock PLL. PLL output to post divider is PPllClk = PPLL_REF * PPLL_FBx_DIV / PPLL_REF_DIV. PPllClk required to always run in the 350 MHz >= PPllClk >= 125 MHz range. For slower pixel clocks us the PPLL_POSTx_DIV field. The PPLL_FBx_DIV setting must be >= 4.
PPLL_ATOMIC_UPDATE_W *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	If atomic update enabled, then controls movement of reference and feedback dividers from pending to active buffers.
PPLL_ATOMIC_UPDATE_R *(mirror of PPLL_REF_DIV:PPLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	Readback of atomic update status.
PPLL_POST3_DIV (DISPLAY)	18:16	0x0	Post divider for pixel clock. See VCLK_SRC_SEL for source of clock that is divided.

Pixel clock PLL feedback and post division selection 3

VCLK_ECP_CNTL - RW - 32 bits - CLKIND:0x8			
Field Name	**Bits**	**Default**	**Description**
VCLK_SRC_SEL	1:0	0x0	Selects source of PIXCLK. The output of this mux selection is post divided by PPLL_POSTx_DIV to create the final PIXCLK. If set to BYTE_CLK, then see BYTE_CLK_POST_DIV below to select the PIXCLK source. Both the clock source you are switching to and from must be running, or the switch will not occur. 0=CPUCLK 1=PSCANCLK 2=BYTE_CLK 3=PPllClk

VCLK_INVERT	4	0x0	Used to invert PPllClk to get opposite duty cycle. Only takes effect when VCLK_SRC_SEL is using PPllClk, and PPLL_POSTx_DIV is divide-by-1. Don't care in other cases. 0=Not Invert 1=Invert
PIXCLK_ALWAYS_ONb	6	0x0	0=PIXCLK is on regardless of CRTC_PIX_WIDTH field (should be set to zero in VGA mode) 1=PIXCLK is off if CRTC_PIX_WIDTH is 0 otherwise PIX-CLK is on
PIXCLK_DAC_ALWAYS_ONb	7	0x0	0=PIXCLK is always on 1=PIXCLK is off during blank time
ECP_DIV	9:8	0x0	OV0CLK clock speed select. Should always be set to 00 for PIXCLK <= 175 MHz and to 01 for PIXCLK > 175 MHz. When set to 01, the overlay and subpic horizontal scale parameters must be adjusted for the extra horizontal replication this causes. Settings 10 and 11 are reserved. 0=VCLK 1=VCLK/2
ECP_FORCE_ON	18	0x0	Controls the dynamic clock control for the back-end overlay/scaler. Set to low for power reduction. 0=SCALER ACTIVITY 1=CONTINUOUS
SUBCLK_FORCE_ON	19	0x0	0=SUBPIC ACTIVITY 1=CONTINUOUS

General controls for the display clocks.
VCLK is the pixel, or dot, clock.
ECP is the overlay/scaler clock.

PIXCLKS_CNTL - RW - 32 bits - CLKIND:0x2D			
Field Name	**Bits**	**Default**	**Description**
PIX2CLK_SRC_SEL	1:0	0x0	0=PIX2CLK_SRC = CPUCLK (input pin) 1=PIX2CLK_SRC = PSCANCLK (input pin) 2=Reserved 3=PIX2CLK_SRC = P2PllClk
PIX2CLK_INVERT	4	0x0	0=Not Invert 1=Invert
PIX2CLK_ALWAYS_ONb	6	0x0	0=PIX2CLK is on regardless of CRTC2_PIX_WIDTH field 1=PIX2CLK is off if CRTC2_PIX_WIDTH is 0 otherwise PIX2CLK is on
PIX2CLK_DAC_ALWAYS_ONb	7	0x0	0=PIX2CLK is always on 1=PIX2CLK is off during blank time

PIXCLK_TV_SRC_SEL	8	0x0	0=PIXCLK_TV = PIXCLK 1=PIXCLK_TV = PIX2CLK
PIXCLK_BLEND_ALWAYS_ONb	11	0x0	0=1 = Enable dynamic stopping for PIXCLK_BLEND
PIXCLK_GV_ALWAYS_ONb	12	0x0	0=1 = Enable dynamic stopping for PIXCLK_GV
PIXCLK_DIG_TMDS_ALWAYS_ONb	13	0x0	
PIXCLK_LVDS_ALWAYS_ONb	14	0x0	
PIXCLK_TMDS_ALWAYS_ONb	15	0x0	

No description available for this register.

PLL_TEST_CNTL - RW - 32 bits - CLKIND:0x13			
Field Name	**Bits**	**Default**	**Description**
TST_SRC_SEL (CG) (CGM)	6:0	0x0	Select different clock source to readable counter and as test clock output
TST_SRC_INV (CG) (CGM)	7	0x0	Invert test clock output
TST_DIVIDERS (CG) (CGM)	8	0x0	Enable driver test
PLL_MASK_READ_B (CG) (CGM)	9	0x1	Mask test clock output and 3 LSBs of readable counter
TESTCLK_MUX_SEL (CG)	12	0x0	0=cg 1=cgm
ANALOG_MON (DISPLAY)	23:15	0x0	
TEST_COUNT (R) (CG) (CGM)	31:24	0x0	Readable counter read back

CG and PLL test control

P2PLL_DIV_0 - RW - 32 bits - CLKIND:0x2C			
Field Name	**Bits**	**Default**	**Description**
P2PLL_FB_DIV (CG)	10:0	0x0	
P2PLL_ATOMIC_UPDATE_W *(mirror of P2PLL_REF_DIV:P2PLL_ATOMIC_UPDATE_W)* (W) (CG)	15	0x0	
P2PLL_ATOMIC_UPDATE_R *(mirror of P2PLL_REF_DIV:P2PLL_ATOMIC_UPDATE_R)* (R) (CG)	15	0x0	
P2PLL_POST_DIV (DISPLAY)	18:16	0x0	0=V2CLK = V2CLK_SRC 1=V2CLK = V2CLK_SRC/2 2=V2CLK = V2CLK_SRC/4 3=V2CLK = V2CLK_SRC/8 4=V2CLK = V2CLK_SRC/3 5=reserved 6=V2CLK = V2CLK_SRC/6 7=V2CLK = V2CLK_SRC/12

No description available for this register.

2.30 TMDS Registers

TMDS_CNTL - RW - 32 bits - [MMReg:0x294]			
Field Name	**Bits**	**Default**	**Description**
TMDS_CTL0	0	0x0	Control signal for TMDS (encoded in Green channel)
TMDS_CTL1	1	0x0	Control signal for TMDS (encoded in Green channel)
TMDS_CTL2	2	0x0	Control signal for TMDS (encoded in Red channel)
TMDS_CTL3	3	0x0	Control signal for TMDS (encoded in Red channel)
TMDS_DEBUG_HSYNC	4	0x0	Debug mode HSYNC control
TMDS_DEBUG_VSYNC	5	0x0	Debug mode VSYNC control
TMDS_DEBUG_DE	6	0x0	Debug mode Display Enable
TMDS_DEBUG_EN	7	0x0	Debug mode enable
TMDS_CTL_FB_SEL	9:8	0x0	Select input of CTL0 for TMDS
TMDS_CTL_FB_DEL	11:10	0x0	Select delay of CTL0 for TMDS
TMDS_STEREOSYNC_CTL_SEL	13:12	0x0	0=TMDS CTL registers have normal functionality 1=Stereosync will use TMDS CTL1 register 2=Stereosync will use TMDS CTL2 register 3=Stereosync will use TMDS CTL3 register
TMDS_SYNC_CHAR_EN	19:16	0x0	SYNC character enable. Each of the bit represents use of register defined sync character. Bit x= 1 means SYNC_CHARx is defined by register
TMDS_SYNC_CONT	24	0x1	TMDS synchronization control
TMDS_DPCUM_TST	25	0x0	Dcbalancer test enable
TMDS_DPCUM_IN	29:26	0x0	Dcbalancer test input
TMDS_CRC_EN	30	0x0	Enable CRC calculation
TMDS_RB_SWITCH_EN	31	0x0	Switch Red and Blue encoding position

Control register for TMDS encoder timing and others

TMDS_SYNC_CHAR_SETA - RW - 32 bits - [MMReg:0x298]			
Field Name	**Bits**	**Default**	**Description**
TMDS_SYNC_CHAR0	9:0	0x354	TMDS SYNC Character 0 (Default 1101010100)
TMDS_SYNC_CHAR1	25:16	0x154	TMDS SYNC Character 1 (Default 0101010100)

TMDS SYNC Character set A

TMDS_SYNC_CHAR_SETB - RW - 32 bits - [MMReg:0x29C]			
Field Name	**Bits**	**Default**	**Description**
TMDS_SYNC_CHAR2	9:0	0xab	TMDS SYNC Character 2 (Default 0010101011)
TMDS_SYNC_CHAR3	25:16	0x2ab	TMDS SYNC Character 3 (Default 1010101011)

TMDS SYNC character set b

TMDS_CRC - R - 32 bits - [MMReg:0x2A0]			
Field Name	**Bits**	**Default**	**Description**
TMDS_CRCRGB	29:0	0x0	30bits RGB combined CRC result

TMDS CRC read back

TMDS_TRANSMITTER_CNTL - RW - 32 bits - [MMReg:0x2A4]			
Field Name	**Bits**	**Default**	**Description**
TMDS_PLLEN	0	0x0	TMDS transmitter's PLL enable. This can power down the PLL
TMDS_PLLRST	1	0x1	TMDS transmitter's PLL reset. PLL will start the locking acquisition process once this signal becomes LOW
TMDS_MODE_SEL	3:2	0x0	
TMDS_REGSEL	5:4	0x0	TMDS_REG output select one of the four channels
TMDS_HALF_CLK_RST	6	0x0	
TMDS_RAN_PAT_RST	7	0x1	
TMDS_TSTPIX	17:8	0x0	TMDS transmitter test pixel. This signals replace pixel value when TSTEN is HIGH
TMDS_REG (R)	27:18	0x0	Output of shift registers. These are the outputs of the 10 shift registers from one of the channels during test mode (Read only)
ICHCSEL	28	0x1	
ITCLKSEL	29	0x0	
TMDS_RAN_PAT_SEL	30	0x0	

TMDS transmitter control register

TMDS_PLL_CNTL - RW - 32 bits - [MMReg:0x2A8]			
Field Name	**Bits**	**Default**	**Description**
TMDS_PLLPCP	2:0	0x4	TMDS PLL charge-pump gain control
TMDS_PLLPVG	5:3	0x4	TMDS PLL VCO gain control
TMDS_PLLPDC	7:6	0x2	TMDS PLL duty cycle control
TMDS_PLLPVS	11:8	0x6	TMDS driver voltage swing control. Select the Rext from 300ohm to 1000ohm in 50ohm step which is equivalent to 250 to 720 mV swing.

TMDS PLL control register

TMDS_PATTERN_GEN_SEED - RW - 32 bits - [MMReg:0x2AC]			
Field Name	**Bits**	**Default**	**Description**
PATTERN_SEED	23:0	0x222222	

No description available for this register.

2.31 LVDS Registers

LVDS_DIGTMDS_CRC - R - 32 bits - [MMReg:0x2CC]			
Field Name	**Bits**	**Default**	**Description**
LVDS_CRC_SIG_RGB	23:0	0x0	CRC signature for red, green and blue components
LVDS_CRC_SIG_C	26:24	0x0	CRC signature for control signals

CRC signature for LVDS or Digital TMDS output

LVDS_GEN_CNTL - RW - 32 bits - [MMReg:0x2D0]			
Field Name	**Bits**	**Default**	**Description**
LVDS_ON	0	0x0	0=LVDS OFF 1=LVDS ON
LVDS_DISPLAY_DIS	1	0x0	0=Normal 1=Force LVDS data to 0
LVDS_PANEL_TYPE	2	0x0	0=TFT single pixel per clock 1=TFT dual pixel per clock
LVDS_PANEL_FORMAT	3	0x1	0=18 bit per pixel TFT panel - 666 RGB 1=24 bit per pixel TFT panel - 888 RGB
LVDS_NO_OF_GREY	5:4	0x0	0=No frame modulation 1=2 levels of grey support 2=4 levels of grey support 3=reserved
LVDS_RST_FM	6	0x0	0=Enable frame modulation circuitry to function 1=Reset frame modulation circuit
LVDS_EN	7	0x0	0=On chip LVDS interface is disabled 1=On chip LVDS interface is enabled
LVDS_BL_MOD_LEVEL	15:8	0x0	
LVDS_BL_MOD_EN	16	0x0	0=Disable LVDS backlight modulation 1=Enable LVDS backlight modulation
LVDS_BL_CLK_SEL	17	0x0	0=29MHz Backlight Modulation Clock 1=29MHz divided by 3 Backlight Modulation Clock
LVDS_DIGON	18	0x0	0=LVDS digital voltage off 1=LVDS digital voltage on
LVDS_BLON	19	0x0	0=LVDS backlight voltage off 1=LVDS backlight voltage on

LVDS_FP_POL	20	0x0	0=active high Frame Pulse / Vsync 1=active low Frame Pulse / Vsync
LVDS_LP_POL	21	0x0	0=active high Line Pulse / Hsync 1=active low Line Pulse / Hsync
LVDS_DTM_POL	22	0x0	0=active high Display Enable / MOD 1=active low Display Enable / MOD
LVDS_SRC_SEL	23	0x0	0=Select Primary (CRTC1) display path 1=Select Secondary (CRTC2) display path
LVDS_RESERVED_BITS	26:24	0x0	
LVDS_FPDI_EN	27	0x1	0=use LDI format for 888 RGB 1=use FPDI format for 888 RGB
LVDS_HSYNC_DELAY	31:28	0x0	

No description available for this register.

LVDS_PLL_CNTL - RW - 32 bits - [MMReg:0x2D4]			
Field Name	**Bits**	**Default**	**Description**
LVDS_CRC_DE_ONLY	0	0x0	0=LVDS CRC includes DE and blank period 1=LVDS CRC includes DE period only
LVDS_LVPVS	4:1	0x6	
LVDS_LPPVG	7:5	0x4	
LVDS_LPPDC	9:8	0x2	
LVDS_LPPCP	12:10	0x4	
LVDS_SS_DISP_EN	13	0x0	0=Disable jitter whole display path 1=Enable jitter whole display path
LVDS_PRG	15:14	0x0	
LVDS_PLL_EN	16	0x0	0=Disable LVDS PLL 1=Enable LVDS PLL
LVDS_PLL_RESET	17	0x0	0=Unreset LVDS PLL 1=Reset LVDS PLL
LVDS_CRC_EN	20	0x0	
LVDS_CRC_UP24	21	0x0	0=Selects lower 30bit (24+syncs,etc) of lvds data as source for LCD2ENG CRC 1=Selects upper 30bit (24+syncs,etc) of lvds data as source for LCD2ENG CRC
LVDS_TEST_DATA_OUT (R)	24:22	0x0	

LVDS_TEST_DATA_SEL	26:25	0x0	
LCDENG_TEST_MODE	31:28	0x0	0000 = Normal 0001 = Eye pattern 1010 = Test mode 0 1011 = Test mode 1 x1xx = Use test clock for LVDS macro

No description available for this register.

LVDS_SS_GEN_CNTL - RW - 32 bits - [MMReg:0x2EC]			
Field Name	**Bits**	**Default**	**Description**
SS_EXT_EN	1	0x0	0=Disable External Spread Spectrum 1=Enable External Spread Spectrum
SS_EXT_SEL	2	0x0	0=Use External Spread Spectrum on Primary Display 1=Use External Spread Spectrum on Secondary Display
SS_BUF_EN	3	0x0	0=Disable Spread Spectrum Buffer 1=Enable Spread Spectrum Buffer
SS_BUF_OFFSET	10:4	0x2	Initial SS buffer read address
SS_BUF_RESET_EN	11	0x0	
SS_BUF_STRENGTH	13:12	0x2	0=Minimum drive. ~74 ohms. 1=Matched impedence drive. ~63 ohms. 2=Optimal drive. ~47 ohms. 3=Maximum drive. ~42 ohms.
LVDSPWR_STRENGTH	15:14	0x2	0=Minimum drive. ~74 ohms. 1=Matched impedence drive. ~63 ohms. 2=Optimal drive. ~47 ohms. 3=Maximum drive. ~42 ohms.
LVDS_PWRSEQ_DELAY1	19:16	0x2	Delay from DIGON active to lvds data active
LVDS_PWRSEQ_DELAY2	23:20	0x3	Delay from lvds data active to BLON active
LVDS_CLK_PATTERN	30:24	0x63	Data pattern for clock channel. Default value is 0x63

Auxiliary window horizontal controls

2.32 DVI Registers

DVI_I2C_CNTL_0 - RW - 32 bits - [MMReg:0x2E0]			
Field Name	**Bits**	**Default**	**Description**
I2C_DONE	0	0x0	0=I2c is busy 1=transfer is complete
I2C_NACK	1	0x0	1=Slave did not issue acknowledge
I2C_HALT	2	0x0	1=Time-out condition, transfer is halted
I2C_SOFT_RST	5	0x0	0=Normal 1=Resets i2c controller
I2C_DRIVE_EN	6	0x0	0=Pullup by external resistor 1=I2C pads drive SDA
I2C_DRIVE_SEL	7	0x0	0=Drive for 10MCLKs 1=20MCLKS
I2C_START	8	0x0	0=No start 1=Start
I2C_STOP	9	0x0	0=No stop 1=Stop
I2C_RECEIVE	10	0x0	0=Send 1=Receive
I2C_ABORT	11	0x0	0=No abort 1=Abort
I2C_GO	12	0x0	
SW_WANTS_TO_USE_DVI_I2C (W)	13	0x0	0=Normal 1=SW requests to use DVI I2C interface
SW_CAN_USE_DVI_I2C (R)	13	0x0	0=DVI I2C interface not available 1=SW has control of the DVI I2C interface
SW_DONE_USING_DVI_I2C (W)	14	0x0	0=if SW has control of DVI I2C, free to use it 1=indicate SW is done using DVI I2C i/f
HW_NEEDS_DVI_I2C (R)	14	0x0	0=no request by H/W for use of DVI I2C 1=H/W HDCP requests use of the DVI I2C i/f
ABORT_HW_DVI_I2C (W)	15	0x0	0=normal operation of H/W using DVI I2C 1=abort current H/W HDCP use of DVI I2C interface
HW_USING_DVI_I2C (R)	15	0x0	0=DVI I2C i/f not in use by H/W 1=HDCP H/W currently using DVI I2C interface
I2C_PRESCALE	31:16	0x0	

No description available for this register.

DVI_I2C_CNTL_1 - RW - 32 bits - [MMReg:0x2E4]			
Field Name	**Bits**	**Default**	**Description**
I2C_DATA_COUNT	3:0	0x0	
I2C_ADDR_COUNT	10:8	0x0	
I2C_SEL	16	0x0	0=Pullup by external resistor 1=I2C pads drive SCL
I2C_EN	17	0x0	
I2C_TIME_LIMIT	31:24	0x0	

No description available for this register.

DVI_I2C_DATA - RW - 32 bits - [MMReg:0x2E8]			
Field Name	**Bits**	**Default**	**Description**
I2C_DATA	7:0	0x0	

No description available for this register.

2.33 flatPanel Registers

FP_H2_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x3C4]			
Field Name	**Bits**	**Default**	**Description**
FP_H2_SYNC_STRT_PIX	2:0	0x0	DFP Horizontal sync start delay in pixels. Allows pixel accurate horizontal positioning by delaying sync position by the specified number of pixels.
FP_H2_SYNC_STRT_CHAR	12:3	0x0	DFP Horizontal sync start position (pixels * 8). Sum of display width, overscan right and front porch. This field is programmed with the exact desired character on which to start HSYNC, not -1. i.e. 656/8 = 82 If horizontal fixed blank or autocentering (see CRTC_MORE_CNTL) is enabled, this register field contains the offset (from then end of the active display/start of horizontal blank) within the blank time of the start of the horizontal sync signal for the DFP (in characters)
FP_H2_SYNC_WID	21:16	0x0	DFP Horizontal sync width (pixels * 8). If horz. fixed blank or autocentering is enabled this register field contains the width (in characters) of the horizontal sync signal for the DFP.
FP_H2_SYNC_POL	23	0x0	DFP Horizontal sync polarity 0=Active high 1=Active low 0=Active high 1=Active low

Secondary display shadow Horizontal Sync for Digital Flat Panels.
Used for DFP HSYNC when FP_GEN_CNTL.DFP_SYNC_SEL=1 and used for CRT HSYNC when FP_GEN_CNTL.CRT_SYNC_SEL=1.
The FP_H_SYNC_STRT_PIX and FP_H_SYNC_POL always apply to the DFP Horizontal Sync regardless of other control bits.(DFP_SYNC_SEL/CRT_SYNC_SEL/horizontal blanking mode/horz. autocentering)

FP_V2_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x3C8]			
Field Name	**Bits**	**Default**	**Description**
FP_V2_SYNC_STRT	11:0	0x0	DFP Vertical sync start position (in lines). Sum of display height, overscan bottom and front porch. If vertical fixed blank or autocentering is enabled(see CRTC_MORE_CNTL) this field contains the offset(from the bottom of the active display/start of vertical blank) within the vertical blank time of the start of the vertical sync signal for the DFP(in lines)

FP_V2_SYNC_WID	20:16	0x0	DFP Vertical sync width(in lines). If vert. fixed blank or autocentering is enabled this field contains the length (in lines) of the vertical sync signal for the DFP
FP_V2_SYNC_POL	23	0x0	DFP Vertical sync polarity 0=Active high 1=Active low 0=Active high 1=Active low

Secondary display shadow Vertical Sync for Digital Flat Panels

FP_GEN_CNTL - RW - 32 bits - [MMReg:0x284]			
Field Name	**Bits**	**Default**	**Description**
FP_ON	0	0x0	0=Resets TMDS block and turns of PIXCLK_TMDS 1=Turns on PIXCLK_TMDS and removes reset to TMDS block
FP_BLANK_EN	1	0x0	0 = Normal operation. 1 = Send black pixel data to TMDS encoder to blank screen. No effect on DE or SYNCs. 0=Normal operation. 1=Black data to TMDS encoder. No effect on DE or SYNCs.
TMDS_EN	2	0x0	0 = TMDS interface is disabled. 1 = TMDS interface is enabled. This bit can be cleared if the hardware detects a hot plug disconnect of the DFP and FP_DETECT_EN=1. 0=TMDS interface is disabled. This bit can be cleared by hot plug disconnect if FP_DETECT_EN=1. 1=TMDS interface is enabled.
PANEL_FORMAT	3	0x1	Panel format: (default=0) 0 = 18 bit per pixel TFT panel - 666 RGB. 1 = 24 bit per pixel TFT panel - 888 RGB 0=18 bit per pixel TFT panel - 666 RGB 1=24 bit per pixel TFT panel - 888 RGB
NO_OF_GREY	5:4	0x0	Grey level select: 0 = No frame modulation. 1 = 2 levels of grey support. 2 = 4 levels of grey support. 3 = reserved 0=No frame modulation. 1=2 levels of grey support. 2=4 levels of grey support. 3=reserved

FP_RST_FM	6	0x0	Frame modulation reset. 0 = EnaFMCir. 1 = ResetFMCir 0=Frame modulator ready. 1=Reset frame modualtor circuit.
FP_EN_TMDS (R)	7	0x0	Read-only bit that indicates TMDS status. 0 = TMDS interface is disabled. Either by TMDS_EN written with 0, or hot plug disconnect detected. 1 = TMDS interface is enabled. Also requires TMDS macro to be properly set up. 0=TMDS interface is disabled. Either by TMDS_EN written with 0, or hot plug disconnect detected. 1=TMDS interface is enabled. Also requires TMDS macro to be properly set up.
FP_DETECT_SENSE (R)	8	0x0	0 = No hot plug panel connected or panel that does not support hot plug is connected. 1 = Hot plug panel is connected. 0=No hot plug panel connected or panel that does not support hot plug is connected. 1=Hot plug panel is connected.
FP_DETECT_INT_POL	9	0x0	0 = Generate interrupt on hot plug disconnect. 1 = Generate interrupt on hot plug connect. Requires interrupt to be enabled(GEN_INT_CNTL.FP_DETECT_MASK) 0=Generate interrupt on hot plug disconnect. 1=Generate interrupt on hot plug connect.
FP_DETECT_EN	12	0x0	0 = Hot plug detection has no affect on TMDS transmitter enable/disable. 1 = Hot plug detection can disable TMDS transmitter. Transmitter enable only possible by SW. 0=Hot plug detection has no affect on TMDS transmitter enable/disable. 1=Hot plug detection can disable TMDS transmitter. Transmitter enable only possible by SW.
FP_SRC_SEL	13	0x0	0=Select Primary (CRTC1) display path 1=Select Secondary (CRTC2) display path
FP_USE_VGA_HVSYNC	14	0x1	0 = TMDS H&V SYNC based only on FP_x_SYNC_STRT_WID registers. For VGA modes, this will not be adjusted for pixel data delays through VGA logic. 1 = TMDS H&V SYNC adjusted in VGA modes for delays through VGA pixel data logic. 0=TMDS H&V SYNC based only on FP_x_SYNC_STRT_WID. For VGA modes, this will not be adjusted for pixel data delays through VGA logic. 1=TMDS H&V SYNC adjusted in VGA modes for delays through VGA pixel data logic.

FP_USE_VGA_SYNC_POLARITY	15	0x0	0 = TMDS H&V SYNC polarities determined by FP_x_SYNC_POL fields. 1 = TMDS H&V SYNC polarities determined by VGA_xSYNC regs in VGA modes, and FP_x_SYNC_POL fields in non-VGA modes. 0=TMDS H&V SYNC polarities determined by FP_x_SYNC_POL fields. 1=TMDS H&V SYNC polarities determined by VGA_xSYNC regs in VGA modes, and FP_x_SYNC_POL fileds in non-VGA modes.
CRTC_DONT_SHADOW_VPAR	16	0x0	0 = Use Shadowed Vertical CRT values. DFP VSYNC calculated from shadow CRT VSYNC register values. If FP_USE_SHADOW_EN=1 uses shadowed V_BLANK_START, V_BLANK_END, and V_TOTAL. 1 = Use non-shadowed vertical CRT values. DFP VSYNC calculated from non-shadowed vertical CRT values. If FP_USE_SHADOW_EN=1, uses non-shadowed V_BLANK_START, V_BLANK_END, and V_TOTAL. 0=Use Shadowed 1=Use non-shadowed
CRTC_DONT_SHADOW_HEND	17	0x0	0 = If FP_USE_SHADOW_EN=1 then use shadowed versions of H_DISP_END, H_BLANK_START, and H_BLANK_END CRT registers 1 = Use non-shadowed CRT register values regardless of value of FP_USE_SHADOW_EN 0=Use Shadowed HorzDispEnd 1=Use non-shadowed HorzDispEnd
CRTC_USE_SHADOWED_VEND	18	0x0	0 = use non-shadow V_DISP_END register value 1 = if FP_USE_SHADOW_EN=1 then use shadow CRT V_DISP_END_S register value, else use non-shadow value 0=disable 1=use shadowed VEND
CRTC_USE_SHADOWED_ROWCUR	19	0x0	0 = use normal CRT register values 1 = use shadowed MAX_ROW_SCAN, CURSOR_START, CURSOR_END, and UNDRLN_LOC CRT registers 0=disable 1=use shadowed RowCur
RMX_HVSYNC_CONTROL_EN	20	0x0	0 = Use standard syncs generated for CRT and DFP HSYNC and VSYNC outputs 1 = Use syncs generated in RMX logic for CRT and DFP HSYNC and VSYNC outputs. While this will also work when RMX is disabled, it should only be used with RMX enabled 0=Use standard syncs generated for CRT and DFP HSYNC and VSYNC outputs 1=Use syncs generated in RMX logic for CRT and DFP HSYNC and VSYNC outputs

DFP_SYNC_SEL	21	0x0	0=Use standard(non-shadow) values for DFP syncs(with auto-scaling panels or at native resolution) 1=Use shadow(DFP) registers for DFP syncs 0=Use non-shadow values for DFP syncs 1=Use shadow(LCD) values for DFP syncs
CRTC_LOCK_8DOT	22	0x0	0=Normal VGA text is 8 or 9 dot depending on SEQ01.SEQ_DOT8. 1=Force VGA text to 8 dot. Used when Horizontal RMX is enabled 0=Normal VGA text 8 or 9 dot. 1=Force VGA text to 8 dot.
CRT_SYNC_SEL	23	0x0	0=Use standard (non-shadow) values for CRT syncs in normal operation. 1=Use shadow(DFP) values for CRT syncs in centring/ RMX modes (registers with FP_ name prefix) 0=Use non-shadow values for CRT syncs in centring mode 1=Use shadow(LCD) values for CRT syncs in centring mode
FP_USE_SHADOW_EN	24	0x0	0=Use standard CRTC registers to calculate CRT display parameters. 1=Use shadow CRT registers (mirrored with the registers with a FP_ name prefix) to calculate CRT parameters. 0=Disable 1=Enable
DONT_RST_CHAREN	25	0x0	0 = stop CHAREN if SEQ00=0 1 = when SEQ00=0 CHAREN is not stopped. Use with RMX. 0=stop CHAREN if SEQ00=0 1=Don't
CRT_SYNC_ALT_SEL	26	0x0	0 = value of CRT_SYNC_SEL determines which register values are used to generate CRT horz. & vert. sync start and width. 1 = Alternate CRT shadow register values (CRT_CRTC_H_SYNC_STRT_WID and CRT_CRTC_V_SYNC_STRT_WID) are used to generate CRT sync start and width in centering/RMX modes. 0=Use CRT_SYNC_SEL to choose CRT sync start and width. 1=Use alternate CRT shadow values for CRT sync start and width in centering mode (CRT_CRTC_H_SYNC_STRT_WID and CRT_CRTC_V_SYNC_STRT_WID)

CRTC_USE_NONSHADOW_HPARAMS_FOR_BLANK	27	0x0	0 = disabled (normal operation) 1 = makes left overscan width in VGA equal the value in non-shadow CRT registers when shadowing is enabled. When FP_USE_SHADOW_EN=1 and in VGA mode, the horizontal blank end(left overscan width) is calculated from the non-shadow CRT values. H_BLANK_END = H_TOTAL(shadow) - (H_TOTAL(non-shadow) - H_BLANK_END(non-shadow)) 0=Left overscan calculated from shadow H parameters. 1=Left overscan calculated from non-shadow H parameters.
CRTC_USE_NONSHADOW_VPARAMS_FOR_BLANK	28	0x0	0 = disabled (normal operation) 1 = makes top overscan width in VGA equal the value in non-shadow CRT registers when shadowing is enabled. When FP_USE_SHADOW_EN=1 and in VGA mode, the vertical blank end(top overscan width) is calculated from the non-shadow CRT values. V_BLANK_END = V_TOTAL(shadow) - (V_TOTAL(non-shadow) - V_BLANK_END(non-shadow)) 0=Top overscan calculated from shadow V parameters. 1=Top overscan calculated from non-shadow V parameters.
CRTC_VGA_XOVERSCAN_COLOR	29	0x0	0 = use extended overscan color (OVR_CLR register) for overscan pixels in VGA mode when extended overscan is enabled(CRTC_VGA_XOVERSCAN = 1). 1 = use VGA overscan color for overscan pixels in VGA mode when extended overscan enabled. 0=Use extended overscan color when doing extended overscan in VGA mode. 1=Use VGA overscan color when doing extended overscan in VGA mode.
CRTC_VGA_XOVERSCAN_DIVBY2_EN	30	0x0	0 = in VGA mode, when CRTC_EXT_CNTL.CRTC_VGA_XOVERSCAN=1 and SEQ_PCLKBY2=1 then OVR_LEFT/RIGHT register values are used. 1 = in VGA mode, when CRTC_EXT_CNTL.CRTC_VGA_XOVERSCAN=1 and SEQ_PCLKBY2=1, then OVR_LEFT/RIGHT register values are divided by 2 0=When VGA and extended overscan and SEQ_PCLKBY2=1, then OVR_WID_LEFT/RIGHT are used directly. 1=When VGA and extended overscan and SEQ_PCLKBY2=1, then OVR_WID_LEFT/RIGHT are divided by 2.

No description available for this register.

FP2_GEN_CNTL - RW - 32 bits - [MMReg:0x288]			
Field Name	**Bits**	**Default**	**Description**
FP2_BLANK_EN	1	0x0	0=Normal operation. 1=Black data to TMDS encoder. No effect on DE or SYNCs.
FP2_ON	2	0x0	
FP2_PANEL_FORMAT	3	0x1	0=18 bit per pixel TFT panel - 666 RGB 1=24 bit per pixel TFT panel - 888 RGB
FP2_NO_OF_GREY	5:4	0x0	0=No frame modulation. 1=2 levels of grey support. 2=4 levels of grey support. 3=reserved
FP2_RST_FM	6	0x0	0=Frame modulator ready. 1=Reset frame modualtor circuit.
FP2_DETECT_SENSE (R)	8	0x0	0=No hot plug panel connected or panel that does not support hot plug is connected. 1=Hot plug panel is connected.
FP2_DETECT_INT_POL	9	0x0	0=Generate interrupt on hot plug disconnect. 1=Generate interrupt on hot plug connect.
FP2_SRC_SEL	13	0x0	0=Select Primary (CRTC1) display path 1=Select Secondary (CRTC2) display path
FP2_FP_POL	16	0x0	0=Active High Frame Pulse 1=Active Low Frame Pulse
FP2_LP_POL	17	0x0	0=Active High Line Pulse 1=Active Low Line Pulse
FP2_SCK_POL	18	0x0	0=Active High Shift Clock 1=Active Low Shift Clock
FP2_LCD_CNTL	21:19	0x0	
FP2_PAD_FLOP_EN	22	0x0	0=To Enable flops in pads.
FP2_CRC_EN	23	0x0	
FP2_CRC_READ_EN	24	0x0	

No description available for this register.

FP_HORZ_STRETCH - RW - 32 bits - [MMReg:0x28C]			
Field Name	**Bits**	**Default**	**Description**
FP_HORZ_STRETCH_RATIO	15:0	0x0	Horizontal RMX Stretch Ratio. When FP_HORZ_STRETCH_MODE = 0 (pixel replication), the value in this register is shifted out serially (LSB first, i.e. bit 0). If the value of the bit shifted out is a '0' then the next real (pre-replication) display pixel is loaded and displayed. If the bit shifted out is a '1' then the last pixel loaded will be duplicated and displayed. The bit shifter is looped back to bit 0 after either bit 9, 11, 12, 14, or 15 depending on the value of FP_LOOP_STRETCH. In pixel-replication mode, bit 0 of the ratio must always be set to '0'. When FP_HORZ_STRETCH_MODE = 1 (blending), valid values for the ratio are 1 to 4095. Ratio = (((Active display width in characters (including overscan)) / (Panel width in characters)) x 4096)
FP_HORZ_PANEL_SIZE	24:16	0x0	Digital Flat Panel (DFP) width in characters = ((panel width in pixels/8) - 1). Used when Automatic RMX Horizontal Ratio enabled
FP_HORZ_STRETCH_EN	25	0x0	0 = Disable Horizontal RMX. 1 = Enable Horz. RMX. 0=Horizontal stretching disabled. 1=Horizontal stretching enabled.
FP_HORZ_STRETCH_MODE	26	0x0	Panel horizontal stretch mode: 0 = pixel replication. 1 = horizontal blending. 0=Disables horizontal filter, giving pixel replication. 1=Enables horizontal filter blending.
FP_AUTO_HORZ_RATIO	27	0x0	0=Use FP_HORZ_STRETCH_RATIO as Hratio for RMX. 1=Use auto-calculated ratio for Hratio. Works in both VGA and Extended modes. 0=Use FP_HORZ_STRECTH_RATIO as Hratio for RMX. Required selection for non-VGA modes. 1=Use auto-calculated ratio for Hratio. Only available in VGA modes.
FP_LOOP_STRETCH	30:28	0x0	Panel horizontal 'pixel replication' loop back select: 0 = after bit 9 loop back to bit 0. 1 = after bit 11 loop back to bit 0. 2 = after bit 12 loop back to bit 0. 3 = after bit 14 loop back to bit 0. 4 = after bit 15 loop back to bit 0 0=Loop bit 0 back to bit 9 1=Loop bit 0 back to bit 11 2=Loop bit 0 back to bit 12 3=Loop bit 0 back to bit 14 4=Loop bit 0 back to bit 15

RMX_AUTO_RATIO_HORZ_INC	31	0x1	0=Horizontal auto ratio result truncated, i.e. rounded down. 1=Horizontal auto ratio result truncated, and then incremented by 1 i.e. rounded up. 0=Horizontal auto ratio result truncated, i.e. rounded down. 1=Horizontal auto ratio result trucated, and then incremented by 1.

Panel Horizontal Ratiometric(RMX) stretch control register

FP_VERT_STRETCH - RW - 32 bits - [MMReg:0x290]			
Field Name	**Bits**	**Default**	**Description**
FP_VERT_STRETCH_RATIO	11:0	0x0	For both values of FP_VERT_STRETCH_MODE, valid values for the ratio are 1 to 4095. Ratio = (((Active display height in lines (including overscan)) / (Panel height in lines)) x 4096)
FP_VERT_PANEL_SIZE	23:12	0x0	DFP height in lines = (panel height in lines - 1)
FP_VERT_STRETCH_EN	25	0x0	0 = Disable Vertical RMX. 1 = Enable Vert. RMX 0=Vertical stretching disabled. 1=Vertical stretching enabled.
FP_VERT_STRETCH_MODE	26	0x0	Panel vertical stretch mode: 0 = line replication. 1 = vertical blending 0=Disables vertical filter, giving line replication. 1=Enables vertical filter blending.
FP_AUTO_VERT_RATIO	27	0x0	0 = Use FP_VERT_STRETCH_RATIO as Vratio for RMX. 1 = use auto-calculated vertical ratio. Works in both VGA and Extended modes. 0=Use FP_VERT_STRETCH_RATIO register field. Required selection for non-VGA modes. 1=Auto calculate vertical ratio. Only availabe in VGA modes.
RMX_AUTO_RATIO_VERT_INC	31	0x1	0 = Vertical auto ratio result truncated(i.e. rounded down). 1 = Vertical auto ratio result truncated, and then incremented by 1 (i.e. rounded up). 0=Vertical auto ratio result truncated, i.e. rounded down. 1=Vertical auto ratio result trucated, and then incremented by 1.

No description available for this register.

FP_H_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x2C4]			
Field Name	**Bits**	**Default**	**Description**
FP_H_SYNC_STRT_PIX	2:0	0x0	DFP Horizontal sync start delay in pixels. Allows pixel accurate horizontal positioning by delaying sync position by the specified number of pixels.
FP_H_SYNC_STRT_CHAR	12:3	0x0	DFP Horizontal sync start position (pixels * 8). Sum of display width, overscan right and front porch. This field is programmed with the exact desired character on which to start HSYNC, not -1. i.e. 656/8 = 82 If horizontal fixed blank or autocentering (see CRTC_MORE_CNTL) is enabled, this register field contains the offset (from then end of the active display/start of horizontal blank) within the blank time of the start of the horizontal sync signal for the DFP (in characters)
FP_H_SYNC_WID	21:16	0x0	DFP Horizontal sync width (pixels * 8). If horz. fixed blank or autocentering is enabled this register field contains the width (in characters) of the horizontal sync signal for the DFP.
FP_H_SYNC_POL	23	0x0	DFP Horizontal sync polarity 0=Active high 1=Active low 0=Active high 1=Active low

Shadow Horizontal Sync for Digital Flat Panels.
Used for DFP HSYNC when FP_GEN_CNTL.DFP_SYNC_SEL=1 and used for CRT HSYNC when FP_GEN_CNTL.CRT_SYNC_SEL=1.
The FP_H_SYNC_STRT_PIX and FP_H_SYNC_POL always apply to the DFP Horizontal Sync regardless of other control bits.(DFP_SYNC_SEL/CRT_SYNC_SEL/horizontal blanking mode/horz. autocentering)

FP_V_SYNC_STRT_WID - RW - 32 bits - [MMReg:0x2C8]			
Field Name	**Bits**	**Default**	**Description**
FP_V_SYNC_STRT	11:0	0x0	DFP Vertical sync start position (in lines). Sum of display height, overscan bottom and front porch. If vertical fixed blank or autocentering is enabled(see CRTC_MORE_CNTL) this field contains the offset(from the bottom of the active display/start of vertical blank) within the vertical blank time of the start of the vertical sync signal for the DFP(in lines)
FP_V_SYNC_WID	20:16	0x0	DFP Vertical sync width(in lines). If vert. fixed blank or autocentering is enabled this field contains the length (in lines) of the vertical sync signal for the DFP

FP_V_SYNC_POL	23	0x0	DFP Vertical sync polarity 0=Active high 1=Active low 0=Active high 1=Active low

Shadow Vertical Sync for Digital Flat Panels

FP_HORZ_VERT_ACTIVE - RW - 32 bits - [MMReg:0x278]			
Field Name	**Bits**	**Default**	**Description**
FP_VERT_ACTIVE_SIZE	11:0	0x0	Vertical Active Display size (usually panel native height) in lines when vert. auto-centering is enabled.
FP_HORZ_ACTIVE_SIZE	24:16	0x0	Horizontal Active Display size (usually panel native width) in characters (8 pixels per character) when horz. auto-centering is enabled.

Auxiliary window horizontal controls

FP_CRTC_H_TOTAL_DISP - RW - 32 bits - [MMReg:0x250]			
Field Name	**Bits**	**Default**	**Description**
FP_CRTC_H_TOTAL	9:0	0x0	Horizontal total ((pixels * 8)-1). Sum of display width, overscan right, front porch, sync width, back porch and overscan left. If horizontal autocentering or horizontal fixed blank is enabled(see CRTC_MORE_CNTL), this field contains the width of the horizontal blanking time(in characters)
FP_CRTC_H_DISP	24:16	0x0	Horizontal display end ((pixels * 8)-1). Determines number of visible pixels, not including overscan.

More CRTC controls
For 'fixed blank' and 'autocentering' modes, refer to registers FP_CRTC_H_TOTAL_DISP, FP_CRTC_V_TOTAL_DISP, FP_H_SYNC_STRT_WID, FP_V_SYNC_STRT_WID, CRT_CRTC_H_SYNC_STRT_WID, and CRT_CRTC_V_SYNC_STRT_WID. For 'autocentering' please also refer to FP_HORZ_VERT_ACTIVE. When autocentering is enabled, fixed blanking is also enabled.

FP_CRTC_V_TOTAL_DISP - RW - 32 bits - [MMReg:0x254]			
Field Name	**Bits**	**Default**	**Description**
FP_CRTC_V_TOTAL	11:0	0x0	Vertical total(lines-1). Sum of display height, overscan bottom, front porch, sync width, back porch and overscan top. If vertical autocentering or vertical fixed blank is enabled(see CRTC_MORE_CNTL) this field contains the length of the vertical blanking time(in lines)
FP_CRTC_V_DISP	27:16	0x0	Vertical display end(lines-1). Determines number of visible lines, not including overscan.

More CRTC controls

For 'fixed blank' and 'autocentering' modes, refer to registers FP_CRTC_H_TOTAL_DISP, FP_CRTC_V_TOTAL_DISP, FP_H_SYNC_STRT_WID, FP_V_SYNC_STRT_WID, CRT_CRTC_H_SYNC_STRT_WID, and CRT_CRTC_V_SYNC_STRT_WID. For 'autocentering' please also refer to FP_HORZ_VERT_ACTIVE. When autocentering is enabled, fixed blanking is also enabled.

2.34 testability Registers

DISP_TEST_MACRO_RW_WRITE - RW - 32 bits - CLKIND:0x1A			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_WRITE1	13:0	0x0	
TEST_MACRO_RW_WRITE2	27:14	0x0	

No description available for this register.

DISP_TEST_MACRO_RW_READ - RW - 32 bits - CLKIND:0x1B			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_READ1	15:0	0x0	
TEST_MACRO_RW_READ2	31:16	0x0	

No description available for this register.

DISP_TEST_MACRO_RW_DATA - R - 32 bits - CLKIND:0x1C			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_DATA	31:0	0x0	

No description available for this register.

DISP_TEST_MACRO_RW_CNTL - RW - 32 bits - CLKIND:0x1D			
Field Name	**Bits**	**Default**	**Description**
TEST_MACRO_RW_START	0	0x0	
TEST_MACRO_RW_OP	3:1	0x0	
TEST_MACRO_RW_MODE	5:4	0x0	
TEST_MACRO_RW_MISMATCH_SEL	14:6	0x0	
TEST_MACRO_RW_MISMATCH (R)	15	0x0	
TEST_MACRO_RW_ENABLE	16	0x0	

TEST_MACRO_RW_SCLK_NEG_ENABLE	17	0x0	

No description available for this register.

2.35 palette Registers

PALETTE_INDEX - RW - 32 bits - [IOReg,MMReg:0xB0]			
Field Name	**Bits**	**Default**	**Description**
PALETTE_W_INDEX	7:0	0x0	Write: Sets starting index for palette writes. Auto-increments on each write to PALETTE_DATA. Read: Indicates index where next write to PALETTE_DATA will be written.
PALETTE_R_INDEX	23:16	0x0	Write: Sets starting index for palette reads. Auto-increments on each read from PALETTE_DATA. Read: Indicates index where next read from PALETTE_DATA will be read.

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

PALETTE_DATA - RW - 32 bits - [IOReg,MMReg:0xB4]			
Field Name	**Bits**	**Default**	**Description**
PALETTE_DATA_B	7:0	0x0	Blue palette data.
PALETTE_DATA_G	15:8	0x0	Green palette data.
PALETTE_DATA_R	23:16	0x0	Red palette data.

Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette. If both bytes are writtten at once, the write index will win and read index will be ignored. Note there is only one internal index register, so writing to one of them changes the read for both.

PALETTE_30_DATA - RW - 32 bits - [IOReg,MMReg:0xB8]			
Field Name	**Bits**	**Default**	**Description**
PALETTE_DATA_B	9:0	0x0	Blue palette data.
PALETTE_DATA_G	19:10	0x0	Green palette data.
PALETTE_DATA_R	29:20	0x0	Red palette data.

Display palette 30 bit data read/write. Use register instead of PALETTE_DATA to take advantage of the 30 bit DACs and improved gamma correction.

2.36 rmx Registers

RMX_HORZ_PHASE - RW - 32 bits - [MMReg:0xDBC]			
Field Name	**Bits**	**Default**	**Description**
RMX_HORZ_START_PHASE	11:0	0x0	Ratiometric expander horizontal accumulator start phase. Horizontal filter accumulator resets to this value at the start of each line. This is normally set to zero, but can be used to fix certain large expansion cases where the expanded image otherwise comes out one pixel too large.
RMX_VERT_START_PHASE	27:16	0x0	Ratiometric expander vertical accumulator start phase. Vertical filter accumulator resets to this value at the start of each frame. This is normally set to zero, but can be used to fix certain large expansion cases where the expanded image otherwise comes out one pixel too large.

Ratiometric expander accumulator horizontal and vertical start phases

2.37 auxWin Registers

AUX_WINDOW_HORZ_CNTL - RW - 32 bits - [MMReg:0x2D8]			
Field Name	**Bits**	**Default**	**Description**
AUX_WIN_HORZ_START	11:0	0x0	Pixel coordinate of left edge of auxiliary window.
AUX_WIN_HORZ_END	23:12	0x0	Pixel coordinate of right edge of auxiliary window.
AUX_WIN_EN	24	0x0	Auxiliary window enable/disable. 0=Disable 1=Enable
AUX_WIN_OUT_DELAY	28:25	0x5	Pixel accurate adjustment of AUXWIN position. Allows it to delayed relative to the RGB signals from the DAC. Available settings are: 000 = -5 pixels 001 = -4 pixels 010 = -3 pixels 011 = -2 pixels 100 = -1 pixel 101 = 0 pixels 110 = +1 pixels 111 = +2 pixels
RESERVED	30	0x1	Not used in this ASIC
AUX_WIN_LOCK *(mirror of AUX_WINDOW_VERT_CNTL:AUX_WIN_LOCK)*	31	0x0	Locks AUX_WIN register fields to allow atomic update of all fields at once when unlocked. 0 = not locked, 1 = locked 0=don't lock 1=lock

Auxiliary window horizontal controls

AUX_WINDOW_VERT_CNTL - RW - 32 bits - [MMReg:0x2DC]			
Field Name	**Bits**	**Default**	**Description**
AUX_WIN_VERT_START	11:0	0x0	Line coordinate of top edge of auxiliary window.
AUX_WIN_VERT_END	27:16	0x0	Line coordinate of bottom edge of auxiliary window.
AUX_WINDOW_POL	30	0x0	Selects whether AUXWIN output is high or low when raster inside window coordinates. 0 = active high 1 = active low 0=Active high 1=Active low

AUX_WIN_LOCK	31	0x0	Locks AUX_WIN register fields to allow atomic update of all fields at once when unlocked. 0 = not locked, 1 = locked 0=don't lock 1=lock

Auxiliary window vertical controls

2.38 snapshot Registers

SNAPSHOT_VH_COUNTS - R - 32 bits - [MMReg:0x240]			
Field Name	**Bits**	**Default**	**Description**
SNAPSHOT_HCOUNT	9:0	0x0	
SNAPSHOT_VCOUNT	27:16	0x0	

Graphics & cursor base address

SNAPSHOT_F_COUNT - R - 32 bits - [MMReg:0x244]			
Field Name	**Bits**	**Default**	**Description**
SNAPSHOT_F_COUNT	20:0	0x0	

No description available for this register.

N_VIF_COUNT - RW - 32 bits - [MMReg:0x248]			
Field Name	**Bits**	**Default**	**Description**
N_VIF_COUNT_VAL	9:0	0x0	
GENLOCK_SOURCE_SEL	31	0x0	

No description available for this register.

SNAPSHOT_VIF_COUNT - RW - 32 bits - [MMReg:0x24C]			
Field Name	**Bits**	**Default**	**Description**
LSNAPSHOT_VIF_COUNT (R)	9:0	0x0	
USNAPSHOT_VIF_COUNT (R)	20:10	0x0	
AUTO_SNAPSHOT_TAKEN_RD (R)	24	0x0	
AUTO_SNAPSHOT_TAKEN_WR (W)	24	0x0	
MANUAL_SNAPSHOT_NOW	25	0x0	

No description available for this register.

SNAPSHOT2_VH_COUNTS - R - 32 bits - [MMReg:0x340]			
Field Name	**Bits**	**Default**	**Description**
SNAPSHOT2_HCOUNT	9:0	0x0	
SNAPSHOT2_VCOUNT	27:16	0x0	

No description available for this register.

SNAPSHOT2_F_COUNT - R - 32 bits - [MMReg:0x344]			
Field Name	**Bits**	**Default**	**Description**
SNAPSHOT2_F_COUNT	20:0	0x0	

No description available for this register.

N_VIF2_COUNT - RW - 32 bits - [MMReg:0x348]			
Field Name	**Bits**	**Default**	**Description**
N_VIF2_COUNT_VAL	9:0	0x0	
GENLOCK2_SOURCE_SEL	31	0x0	

No description available for this register.

SNAPSHOT2_VIF_COUNT - RW - 32 bits - [MMReg:0x34C]			
Field Name	**Bits**	**Default**	**Description**
LSNAPSHOT2_VIF_COUNT (R)	9:0	0x0	
USNAPSHOT2_VIF_COUNT (R)	20:10	0x0	
AUTO_SNAPSHOT2_TAKEN_RD (R)	24	0x0	
AUTO_SNAPSHOT2_TAKEN_WR (W)	24	0x0	
MANUAL_SNAPSHOT2_NOW	25	0x0	

No description available for this register.

2.39 tvout Registers

TV_MASTER_CNTL - RW - 32 bits - [MMReg:0x800]			
Field Name	**Bits**	**Default**	**Description**
TV_ASYNC_RST	0	0x1	
CRT_ASYNC_RST	1	0x1	
RESTART_PHASE_FIX	3	0x0	
TV_FIFO_ASYNC_RST	4	0x1	
reserved	5		reserved
EXTRA_BIT_ONE_0	6	0x1	
CRT_FIFO_CE_EN	9	0x1	0=Operation enable of crt fifo off during reset, on otherwise 1=Operation enable of crt fifo off during reset and horizontal blank, on otherwise
TV_FIFO_CE_EN	10	0x1	0=Operation enable of tv fifos always on 1=Operation enable of tv fifos on only when reads/writes are being requested
RE_SYNC_NOW_SEL	15:14	0x0	0=Use Horizontal sync to generate RE_SYNC_NOW 2=Use Shadow Horizontal sync to generate RE_SYNC_NOW 3=Use Horizontal blank to generate RE_SYNC_NOW
EXTRA_BIT_ZERO_1	16	0x0	
EXTRA_BIT_ONE_1	17	0x1	
EXTRA_BIT_ZERO_2	18	0x0	
EXTRA_BIT_ONE_2	19	0x1	
TVCLK_ALWAYS_ONb	30	0x0	0=TVCLK is on regardless of the TV_ON bit 1=TVCLK is off if TV_ON is 0 otherwise TVCLK is on
TV_ON	31	0x0	0=TV Off 1=TV On

No description available for this register.

TV_RGB_CNTL - RW - 32 bits - [MMReg:0x804]			
Field Name	**Bits**	**Default**	**Description**
UV_DITHER_EN	3:2	0x0	UV dither enable 0=Enable error defused dithering. 1=Disable dithering by rounding. 2=Enable positional dithering. 3=Disable dithering by truncating.
SWITCH_TO_BLUE	4	0x1	Setting to 1 replaces input video data with blue screen. 0=Pass RGB data to RGBtoYUV converter. 1=Force R to 0, G to 0, and B = 255.
RGB_DITHER_EN	5	0x0	0=Disable 1=Enable
RGB_SRC_SEL	9:8	0x0	RGB input source select 0=Select primary display before ratiometric expansion 1=Select primary display after ratiometric expansion 2=Select secondary display before ratiometric expansion 3=Reserved
RGB_CONVERT_BY_PASS	10	0x0	Conversion for RGB Input Source Path 0=Convert RGB to YUV 1=Bypass Conversion
RE_SYNC_NOW_POS	11	0x0	RE_SYNC_NOW position. Note: When '1', D_HDISP and D_VDISP must be programmed to include the TOP, BOT, LEFT and RIGHT overscan as if this is part of the display data. 0=Position RE_SYNC_NOW at line VDISP+1 (normal) 1=Position RE_SYNC_NOW at line VDISP+OVR_BOT+1 (D_HDISP and D_VDISP must be programmed to include TOP,BOT,LEFT and RIGHT overscan as if it is part of the display data
IGNORE_ODD_LINES	13	0x0	Causes D_HBLANK to be active during odd lines 0=Normal operation 1=D_HBLANK is active during odd display lines causing data to be ignored by vscaler
DSP_VCOUNT_RST	14	0x0	Resets vertical count in primary display to CRTC_V_DISP for testing 0=Normal 1=Resets vertical count in primary display to CRTC_V_DISP - for testing
DSP2_VCOUNT_RST	15	0x0	Resets vertical count in secondary display to CRTC2_VDISP for testing 0=Normal 1=Resets vertical count in secondary display to CRTC2_V_DISP - for testing
UVRAM_READ_MARGIN	19:16	0xb	
FIFORAM_FFMACRO_READ_MARGIN	23:20	0x7	

RGB_ATTEN_SEL	25:24	0x0	0=No Attenuation of RGB data 1=Use RGB_ATTEN_VAL to Attenuate Graphics Data Only 2=Use RGB_ATTEN_VAL to Attenuate Video Data Only 3=Use RGB_ATTEN_VAL to Attenuate both Graphics and Video Data
RGB_ATTEN_VAL	31:28	0x0	Reduce brightness by the selected factor

No description available for this register.

TV_SYNC_CNTL - RW - 32 bits - [MMReg:0x808]			
Field Name	**Bits**	**Default**	**Description**
SYNC_OE	0	0x0	SYNC Output Enable 0=tristate output buffer 1=enable output buffer
SYNC_OUT	1	0x0	SYNC Output Data
SYNC_IN (R)	2	0x0	SYNC Input Data (Read only)
SYNC_PUB	3	0x1	SYNC Pull-up Enable (active low) 0=Enable pullup 1=Off
SYNC_PD	4	0x0	SYNC Pull-down Enable (active high) 0=Off 1=Enable pulldown
TV_SYNC_IO_DRIVE	5	0x1	0=Do not boost SYNC drive strength 1=Boost sync drive strength
SYNC_MX	11:8	0x0	SYNC Mux 0=iSYNC_OUT 1=iCOMP_SYNC 2=iHSYNCb 3=iVSYNCb 4=iH_ACTIVE 5=iV_ACTIVE 6=iV_EVENT 7=iY_PIX_EN(0) 8=iY_PIX_EN(1) 9=DacR(8) 10=iDBG_AGC_AMPL_SEL(0) 11=DBG_AGC_AMPL_SEL(1) 12=iDBG_AGC_PULS_MODE 13=iDBG_FIFOARB_REQH(0) 14=iDBG_FIFOARB_REQL(0) 15=iDBG_FIFOARB_GNT(0)

No description available for this register.

TV_HTOTAL - RW - 32 bits - [MMReg:0x80C]			
Field Name	**Bits**	**Default**	**Description**
D_HTOTAL	10:0	0x32f	Program with the number of clocks in a display line minus 1. This value should be (graphics chip H_TOTAL register)*8 + (graphics chip TVO_CNTL register, TVO_H_TOT_PIX + 7). Default is 815 which supports 640x480 resolution.

No description available for this register.

TV_HDISP - RW - 32 bits - [MMReg:0x810]			
Field Name	**Bits**	**Default**	**Description**
D_HDISP	9:0	0x27f	This value is the number of active pixels in a line minus 1. For accelerator modes it should be (graphics chip H_DISP register *8) + 7. However for VGA modes, the above formula does not necessarily apply. For example, keep in mind that there may be 8 or 9 pixels per character clock in VGA modes.

No description available for this register.

TV_HSTART - RW - 32 bits - [MMReg:0x818]			
Field Name	**Bits**	**Default**	**Description**
D_HSTART	10:0	0x28b	This value is the value that the Shadow CRTC D_HCount assumes when a RE_SYNC_NOW from the display CRTC is detected.

No description available for this register.

TV_HCOUNT - R - 32 bits - [MMReg:0x81C]			
Field Name	**Bits**	**Default**	**Description**
D_HCOUNT	10:0	0x0	This value is the Shadow CRTC D_HCount. It is read only.

No description available for this register.

TV_VTOTAL - RW - 32 bits - [MMReg:0x820]			
Field Name	**Bits**	**Default**	**Description**
D_VTOTAL	9:0	0x24d	Program with the number of display lines in a frame minus 1. This value should be the same as the graphics chip V_TOTAL register. Default is 589 which supports 640x480 resolution.

No description available for this register.

TV_VDISP - RW - 32 bits - [MMReg:0x824]			
Field Name	**Bits**	**Default**	**Description**
D_VDISP	9:0	0x1df	This value is the number of active lines minus 1.

No description available for this register.

TV_VCOUNT - R - 32 bits - [MMReg:0x828]			
Field Name	**Bits**	**Default**	**Description**
D_VCOUNT	9:0	0x0	This value is the Shadow CRTC D_VCount. It is read only.

No description available for this register.

TV_FTOTAL - RW - 32 bits - [MMReg:0x82C]			
Field Name	**Bits**	**Default**	**Description**
D_FTOTAL	3:0	0x1	This value should be programmed to equal the number of CRT frames that occur during one complete cycle of the TV timing minus 1. For example, standard NTSC timing has a cycle involving 4 fields, so we program 3 for this type of timing. NTSC with frozen dot crawl requires only 2 fields, so we program 1.

No description available for this register.

TV_FCOUNT - R - 32 bits - [MMReg:0x830]			
Field Name	**Bits**	**Default**	**Description**
D_FCOUNT	3:0	0x0	This value is the Shadow CRTC D_FCOUNT (Frame Count). It is read only.

No description available for this register.

TV_FRESTART - RW - 32 bits - [MMReg:0x834]			
Field Name	**Bits**	**Default**	**Description**
D_FRESTART	3:0	0x0	Indicates the frame during which a restart event will be generated. The restart will cause a restart of TV timing.

No description available for this register.

TV_HRESTART - RW - 32 bits - [MMReg:0x838]			
Field Name	**Bits**	**Default**	**Description**
D_HRESTART	10:0	0x296	Indicates the pixel during which a restart event will be generated. The restart will cause a restart of the TV timing.

No description available for this register.

TV_VRESTART - RW - 32 bits - [MMReg:0x83C]			
Field Name	**Bits**	**Default**	**Description**
D_VRESTART	9:0	0x242	Indicates the line during which a restart event will be generated. The restart will cause a restart of the TV timing.

No description available for this register.

TV_HOST_READ_DATA - R - 32 bits - [MMReg:0x840]			
Field Name	**Bits**	**Default**	**Description**
HOST_RD_DATA_W0	15:0	0x0	Read Data for HOST FIFO when HOST_FIFO_RD is set to ‘1’ and then HOST_FIFO_RD_ACK is polled ‘1’.
HOST_RD_DATA_W1	27:16	0x0	See above

No description available for this register.

TV_HOST_WRITE_DATA - RW - 32 bits - [MMReg:0x844]			
Field Name	**Bits**	**Default**	**Description**
HOST_WT_DATA_W0	15:0	0x0	Write Data for HOST FIFO, depending on HOST_FIFO_WT.
HOST_WT_DATA_W1	27:16	0x0	See above

No description available for this register.

TV_HOST_RD_WT_CNTL - RW - 32 bits - [MMReg:0x848]			
Field Name	**Bits**	**Default**	**Description**
HOST_ADR	8:0	0x0	
HOST_FIFO_RD	12	0x0	0=read cycle inactive/complete (only complete when HOST_FIFO_RD_ACK is 1) 1=read cycle requested
HOST_FIFO_RD_ACK (R)	13	0x0	0=inactive (when no read request) or waiting for read cycle to complete (when read request initiated by HOST_FIFO_RD). 1=read cycle complete
HOST_FIFO_WT	14	0x0	0=write cycle inactive/complete (only complete when HOST_FIFO_WT_ACK is 1) 1=write cycle requested
HOST_FIFO_WT_ACK (R)	15	0x0	0=inactive (when no write request) or waiting for write cycle to complete (when write request initiated by HOST_FIFO_WT). 1=write cycle complete

No description available for this register.

TV_VSCALER_CNTL1 - RW - 32 bits - [MMReg:0x84C]			
Field Name	**Bits**	**Default**	**Description**
UV_INC	15:0	0x2b	This value controls the vertical scaling of UV data of the TV image. Because CRT and TV must have the same percentage of Vertical Blank, this value must be calculated from V_TOTAL and a knowledge of the number of lines in the TV timing. For example, UV_INC = (D_VTOTAL+1)*(1<<0xe)*2/525)). Incorrect setting of this register will cause FIFO underflow or overflow which will shift the colour with respect to the Y image.

UV_THINNER	22:16	0x0	This value is used to control the number of source lines in which the UV data are not blended to create a line on the TV. Shift this 7 bit value up by 9 bits and subtract it from the UV_INC value. The resulting value indicates how many lines are blended. There is a restriction on the maximum number of lines to be excluded from blending, which is (UV_INC - 1.0) lines. If the UV_THINNER value is 0, then the maximum amount of line averaging is done to reduce flicker. A larger value results in less averaging, and will increase the vertical sharpness in exchange for more flicker. This is a 2.5 fixed point value with 5 decimal places. 0x20 - when non-scaling(CRTCVtotal=524 in NTSC, 624 in PAL), tv-pass through. 0x10 - max UV flicker removal for other modes
Y_W_EN	24	0x1	This bit will prevent the VScaler from writing Y data to the FIFO. It can be used to prevent the VScaler from overwriting test patterns that are downloaded into the FIFO directly. Default is 1 (writes enabled).
Y_DEL_W_SIG	28:26	0x2	These bits are used alter the setup time when writing data to FIFOArb from Vscaler. 0=a Dword to FIFOArb will be ready three PIX_CLK later 1=a Dword to FIFOArb will be ready two PIX_CLK later 2=a Dword to FIFOArb will be ready one PIX_CLK later 3=a Dword to FIFOArb is ready (when PIX_CLK is slower than 55% of TV_CLK) 4=a Dword to FIFOArb will be ready four PIX_CLK later
RESTART_FIELD	29	0x1	Vscaler restart field - 0 restart on field 1, 1 - restart on field 0

No description available for this register.

TV_TIMING_CNTL - RW - 32 bits - [MMReg:0x850]			
Field Name	**Bits**	**Default**	**Description**
H_INC	11:0	0x0	
REQ_DELAY	18:16	0x2	0=2 cycle after first modulator requester 1=3 cycle after first modulator requester 2=4 cycle after first modulator requester 3=5 cycle after first modulator requester 4=6 cycle after first modulator requester 5=7 cycle after first modulator requester 6=8 cycle after first modulator requester 7=9 cycle after first modulator requester
REQ_Y_FIRST	19	0x1	0=Request UV data first, and Y data second 1=Request Y data first, and UV data second
reserved	20		reserved

FORCE_BURST_ALWAYS	21	0x0	
UV_POST_SCALE_BYPASS	23	0x0	0=Enable UV post scale multiplier 1=Bypass UV post scale multiplier
UV_OUTPUT_POST_SCALE	31:24	0x72	

No description available for this register.

TV_VSCALER_CNTL2 - RW - 32 bits - [MMReg:0x854]			
Field Name	**Bits**	**Default**	**Description**
DITHER_MODE	0	0x0	Dither mode 0=Randomn defused dithering 1=HcountLSB xor VcountLSB positional dithering
Y_OUTPUT_DITHER_EN	1	0x0	Y output dither enable 0=Disable dithering 1=Enable dithering mode as set by DITHER_MODE
UV_OUTPUT_DITHER_EN	2	0x0	UV output dither enable 0=Disable dithering 1=Enable dithering mode as set by DITHER_MODE
UV_TO_BUF_DITHER_EN	3	0x0	UV to buffer dither enable 0=Disable dithering 1=Enable dithering mode as set by DITHER_MODE
UV_ACCUM_INIT	31:24	0x10	These bits are used to initialize the UV Accumulator in Vscaler between TV frames. This is a 2.6 fixed point value with 6 decimal places, default 0x10=0.25 decimal

No description available for this register.

TV_Y_FALL_CNTL - RW - 32 bits - [MMReg:0x858]			
Field Name	**Bits**	**Default**	**Description**
Y_FALL_ACCUM_INIT	15:0	0x10	This value is used to initialize the falling wave of the Y filters.It is calculated as follows. This is a 3.13 fixed point value with 13 decimal places.Y_FALL_ACCUM_INIT =UV_ACCUM_INIT*Y_SAW_TOOTH_SLOPE
Y_FALL_PING_PONG	16	0x1	This bit assigns the first falling wave to one of the Y filters. 0=assign the first falling wave to Y0 filter 1=assign the first falling wave to Y1 filter

Y_COEF_EN	17	0x0	This bit enables the flat Y filters. 0=set Y filters to the triangle filter; when Y flicker removal = 2, 4 lines 1=set Y filters to the flat filter; when Y flicker removal = 3, 5, 6 lines
Y_COEF_VALUE	31:24	0x0	This value sets the flat Y filter value. This is a 0.8 fixed point value with 8 decimal places.0 - when Y flicker removal =2, 4 lines;128 (or 0x80) - when Y flicker removal =3 lines; 64 (or 0x40) - when Y flicker removal = 5, 6 lines.

No description available for this register.

TV_Y_RISE_CNTL - RW - 32 bits - [MMReg:0x85C]			
Field Name	**Bits**	**Default**	**Description**
Y_RISE_ACCUM_INIT	15:0	0x10	This value is used to initialize the rising wave of the Y filters.It is calculated as follows. This is a 3.13 fixed point value.when (YFlickerRemoval - UV_INC) < UV_ACCUM_INITY_RISE_ACCUM_INIT = (YFlickerRemoval - V_ACCUM_INIT)*Y_SAW_TOOTH_SLOPE;elseY_RISE_ACCUM_INIT = (YFlickerRemoval - V_ACCUM_INIT - UV_ACCUM_INIT) * Y_SAW_TOOTH_SLOPE
Y_RISE_PING_PONG	16	0x0	This bit assigns the first rising wave to one of the Y filters. 0=assign the first rising wave to Y0 filter 1=assign the first riseing wave to Y1 filter
reserved	17		reserved
reserved	18		reserved
reserved	20		reserved
reserved	21		reserved
reserved	22		reserved

No description available for this register.

TV_Y_SAW_TOOTH_CNTL - RW - 32 bits - [MMReg:0x860]			
Field Name	**Bits**	**Default**	**Description**
Y_SAW_TOOTH_AMP	15:0	0x10	This value is used to set max amplitude of the rising and falling waves.It is calculated as follows. This is a 3.13 fixed point value.Y_SAW_TOOTH_AMP = UV_INC * Y_SAW_TOOTH_SLOPE

Y_SAW_TOOTH_SLOPE	31:16	0x10	This value is used to set the slope of y filters. This is a 3.13 fixed value.0x2000 - set slope to 1 when Y Flicker Removal is 2 (tv-passthrough mode)0x1000 - set slope to 1/2 when Y Flicker Removal is 3 (CRTCVtotal < 524 in NTSC)0x0800 - set slope to 1/4 when Y Flicker Removal is 4 & 5;0x0400 - set slope to 1/8 when Y Flicker Removal is 6;(default = 0x0800)UV_INC <= Y Flicker Removal < 2 * UV_INC

No description available for this register.

TV_UPSAMP_AND_GAIN_CNTL - RW - 32 bits - [MMReg:0x864]			
Field Name	**Bits**	**Default**	**Description**
YUPSAMP_EN	0	0x1	Enable Y upsampling filter 0=Disable 1=Enable
YUPSAMP_FLAT	1	0x0	Force Y upsampling filter to use centre tap 0=Use Coefficients 1=Take Centre Tap
UVUPSAMP_EN	2	0x1	Enable U,V upsampling filters 0=Disable 1=Enable
UVUPSAMP_FLAT	3	0x0	Force U,V upsampling filters to use centre tap 0=Use Coefficients 1=Take Centre Tap
YUPSAMP_WINDOW_CNTL	4	0x0	0=Wide window 1=Narrow window
YUPSAMP_COEF_SEL	5	0x0	0=Text 1=4MHz
Y_BREAK_EN	8	0x0	Enable Y gain break: The break point is specified by register field Y_GAIN_LIMIT in GAIN_LIMIT_SETTINGS 0=Disable 1=Enable
UV_BREAK_EN	10	0x0	Enable U,V gain break: The break point is specified by register field UV_GAIN_LIMIT in GAIN_LIMIT_SETTINGS 0=Disable 1=Enable

Controls for Y,U,V upsampling and gain control

TV_GAIN_LIMIT_SETTINGS - RW - 32 bits - [MMReg:0x868]			
Field Name	**Bits**	**Default**	**Description**
Y_GAIN_LIMIT	10:0	0x5ff	
UV_GAIN_LIMIT	24:16	0x17f	

No description available for this register.

TV_LINEAR_GAIN_SETTINGS - RW - 32 bits - [MMReg:0x86C]			
Field Name	**Bits**	**Default**	**Description**
Y_GAIN	8:0	0x100	Unsigned 1.8 bit gain value for the luminanace (Y) portion of the video signal.
UV_GAIN	24:16	0x100	Unsigned 1.8 bit gain setting for the chrominance (U and V) portions of the video signal.

This register contains 9 bit gain values. The fields are in unsigned 1.8 bit format, i.e. 100000000 represents a gain of 1.0. The maximum gain is 1.125 (9/8). Values above the maximum will be limited to 1.125. In the range 7/8 - 9/8, there are eight bits of fractional resolution available, while for values below 7/8 only five bits of fractional resolution are available.

TV_MODULATOR_CNTL1 - RW - 32 bits - [MMReg:0x870]			
Field Name	**Bits**	**Default**	**Description**
YY_FILT_BLEND	1:0	0x2	Alpha blend control for S-Video output: 00=> 100% S-Video filter, 01=> 75% S-Video filter, 25% Composite, 10=>% S-Video filter,% Composite, 11=> 25% S-Video filter, 75% Composite.
YFLT_EN	2	0x1	Composite, and SVID Y-filter enable. 0=Bypass Composite and SVID Y-filters 1=Y-filters enabled
UVFLT_EN	3	0x1	U and V filter enable. 0=Bypass U and V filters 1=Enable U and V filters
ALT_PHASE_EN	6	0x0	Phase alternating line enable. 0=Disabled. (NTSC) 1=Enabled. (PAL)
SYNC_TIP_LEVEL	7	0x0	Composite and Y sync tip level 0=Sets the output sync tip to 0 1=Sets the output sync tip to 4

SET_UP_LEVEL	14:8	0x0	Video setup level, unsigned 7-bit register. For NTSC video format: SET_UP_LEVEL = (4 * SYNC_TIP_LEVEL)+((40IRE + 7.5IRE) * 128/92.5IRE)For PAL video format: SET_UP_LEVEL = (4 * SYNC_TIP_LEVEL)+(43IRE * 128/100IRE)For SCART video format: SET_UP_LEVEL = 0
BLANK_LEVEL	22:16	0x0	Video blank level, unsigned 7-bit register. For NTSC video format: BLANK_LEVEL = (4 * SYNC_TIP_LEVEL)+(40IRE * 128/92.5IRE)For PAL video format: BLANK_LEVEL = (4 * SYNC_TIP_LEVEL)+(43IRE * 128/100IRE)For SCART video format: BLANK_LEVEL = 0
SLEW_RATE_LIMIT	23	0x1	Disable/enable slew rate limiting filter during blanking. 0=No slew rate limit 1=Limit slew rate during blank
FORCE_BLACK_WHITE	24	0x0	Force black&white video. 0=colour ON 1=colour OFF
reserved	26		reserved
reserved	27		reserved
CY_FILT_BLEND	31:28	0x6	Control blending filters for composite Y output. Bits (3:2) select the alternate filter: 00=> Composite, 01=> S-Video, 10=> 1:1 Flat, 11=> Centre Top. Bits (1:0) Alpha blend control for Composite output: 00=> 100% alternate filter, 01=> 75% alternate, 25% composite, 10=>50% alternate,50% composite, 11=> 25% alternate, 75% composite.

No description available for this register.

TV_MODULATOR_CNTL2 - RW - 32 bits - [MMReg:0x874]			
Field Name	**Bits**	**Default**	**Description**
U_BURST_LEVEL	8:0	0x0	U burst level, signed 9-bit register. For NTSC video formats: U_BURST_LEVEL = (-20IRE * 512)/92.5IRE = -111For PAL video formats: U_BURST_LEVEL = 0.707*(-21.5IRE * 512)/100IRE = -78For SCART video format: U_BURST_LEVEL = 0
V_BURST_LEVEL	24:16	0x0	V burst level, signed 9-bit register. For NTSC video formats: V_BURST_LEVEL = 0For PAL video formats: V_BURST_LEVEL = (0.707*(21.5IRE * 512)/100IRE)/1.25 = 62For SCART video format: V_BURST_LEVEL = 0
SETUP_DELAY	31:26	0x0	

No description available for this register.

TV_PRE_DAC_MUX_CNTL - RW - 32 bits - [MMReg:0x888]			
Field Name	**Bits**	**Default**	**Description**
Y_RED_EN	0	0x0	Enables the Y/Red gun of the DAC
C_GRN_EN	1	0x0	Enables the C/Green gun of the DAC
CMP_BLU_EN	2	0x0	Enables the Comp/Blue gun of the DAC
DAC_DITHER_EN	3	0x1	Enables the DAC Dither
RED_MX	6:4	0x0	RED Mux 0=TV Out S-Video Y channel (Default) 1=TV Out Red for RGB formats 2=TV Out Y for YUV formats 3=TV Out Green for RGB formats 4=TV Out Blue for RGB formats 5=0 6=FORCE_DAC_DATA (testing of DAC levels) 7=pins iTEST_DEBUG_IN (testing of DAC levels)
GRN_MX	10:8	0x0	GREEN Mux 0=TV Out S-Video C channel (Default) 1=TV Out Green for RGB formats 2=TV Out U for YUV formats 3=TV Out Blue for RGB formats 4=TV Out Red for RGB formats 5=0 6=FORCE_DAC_DATA (testing of DAC levels) 7=pins iTEST_DEBUG_IN (testing of DAC levels)
BLU_MX	14:12	0x0	BLUE Mux 0=TV Out Composite Video (Default) 1=TV Out Blue for RGB formats 2=TV Out V for YUV formats 3=TV Out Red for RGB formats 4=TV Out Green for RGB formats 5=0 6=FORCE_DAC_DATA (testing of DAC levels) 7=pins iTEST_DEBUG_IN (testing of DAC levels)
FORCE_DAC_DATA	25:16	0x0	Data to force at DAC input when 9 is selected on above muxes

Use this register to select which signals will go to the TV DAC. Also, the video in to TV out bypass filter controls are in this register.

TV_DAC_CNTL - RW - 32 bits - [MMReg:0x88C]			
Field Name	**Bits**	**Default**	**Description**
NBLANK	0	0x1	TV DAC blank 0=zeros DAC output to black 1=normal
NHOLD	1	0x1	TV DAC hold 0=holds DAC data on output 1=normal
PEDESTAL	2	0x0	TV DAC pedestal 0=normal 1=7.5IRE
DETECT	4	0x1	TV monitor detect enable 0=disables monitor detect 1=enables monitor detect
CMPOUT (R)	5	0x0	Detects TV monitor when DETECT is enabled; when detect is disabled, CMPOUT=0. 0=no monitor connected 1=monitor connected
BGSLEEP	6	0x0	0=normal 1=powerdown Bandgap
STD	9:8	0x1	TV DAC output standard 0=PAL 1=NTSC 2=PS2 3=RS343
MON	15:12	0x0	TV DAC analog test bits
BGADJ	19:16	0x0	
DACADJ	23:20	0x0	
RDACPD	24	0x0	0=normal 1=powerdown RDAC
GDACPD	25	0x0	0=normal 1=powerdown GDAC
BDACPD	26	0x0	0=normal 1=powerdown BDAC
RDACDET (R)	29	0x0	0=no RDAC monitor connected 1=RDAC monitor connected
GDACDET (R)	30	0x0	0=no GDAC monitor connected 1=GDAC monitor connected
BDACDET (R)	31	0x0	0=no BDAC monitor connected 1=BDAC monitor connected

TV_CRC_CNTL - RW - 32 bits - [MMReg:0x890]			
Field Name	**Bits**	**Default**	**Description**
V_COMP_DATA_EN	1:0	0x0	RGB Data Select 0=RGB 1=R only 2=G only 3=B only
V_COMP_GATE	2	0x0	Video Compactor Accumulation mode 0=Video Compactor gated to TV_RSTRT_TO_VS signal 1=Video Compactor free-run mode
V_COMP_EN	3	0x0	Video Compactor Enable/Reset 0=Reset/Disable 1=Enable
RST_SUBC_ONRSTRT	4	0x0	Reset subcarrier phase counter every restart edge 0=no reset 1=do reset
CRC_TV_RSTRT_SEL	5	0x0	0=TV_RSTRT_TO_VS toggles when tv timing is reset by CRT's RESTART_TV 1=TV_RSTRT_TO_VS toggles when tv timing hits its free running vsync at FIRST_FIELD_PULSEP

No description available for this register.

TV_VIDEO_PORT_SIG - R - 32 bits - [MMReg:0x894]			
Field Name	**Bits**	**Default**	**Description**
CRC_SIG	29:0	0x0	(29 : 20) B - Signature ,(19 :10) G- Signature , (9:0) R - Signature

CRC Signature

TV_VBI_CC_CNTL - RW - 32 bits - [MMReg:0x898]			
Field Name	**Bits**	**Default**	**Description**
VBI_CC_DATA	15:0	0x0	VBI Data for Closed Caption. Note: Location of CC line is determined by timing codes (7bit ascii + 1 bit parity times two characters)
VBI_CC_WT	24	0x0	Initiates a write cycle requested to Closed Caption using VBI_CC_DATA 0=write cycle inactive/complete (only complete when VBI_CC_WT_ACK is ‘1’) 1=write cycle requested

VBI_CC_WT_ACK (R)	25	0x0	Indicates write cycle to Closed Caption is complete 0= inactive (when no write request) or waiting for write cycle to complete (when write request initiated by VBI_CC_WT). 1=write cycle complete
VBI_CC_HOLD	26	0x0	Hold Closed Caption data to value of VBI_CC_DATA 0=normal; each write will produce CC data only once and if no data is written before the CC line, the data will be null 1= every CC line will use VBI_CC_DATA; VBI_CC_DATA is loaded rather than null when the CC line is complete.
VBI_DECODE_EN	31	0x0	Enables Decode of VBI information in timing generator 0=do not decode VBI information in timing codes (back-wards compatible) 1=decode VBI information in timing codes

No description available for this register.

TV_VBI_EDS_CNTL - RW - 32 bits - [MMReg:0x89C]			
Field Name	**Bits**	**Default**	**Description**
VBI_EDS_DATA	15:0	0x0	VBI Data for Extended Data Service (EDS/XDS). Note: Location of EDS/XDS line is determined by timing codes
VBI_EDS_WT	24	0x0	Initiates a write cycle requested to EDS using VBI_EDS_DATA 0=write cycle inactive/complete (only complete when VBI_EDS_WT_ACK is ‘1’) 1=write cycle requested
VBI_EDS_WT_ACK (R)	25	0x0	Indicates write cycle to EDS is complete 0=inactive (when no write request) or waiting for write cycle to complete (when write request initiated by VBI_EDS_WT). 1=write cycle complete
VBI_EDS_HOLD	26	0x0	Hold EDS data to value of VBI_EDS_DATA 0=normal; each write will produce EDS data only once and if no data is written before the EDS line, the data will be null 1=every EDS line will use VBI_EDS_DATA; VBI_EDS_DATA is loaded rather than null when the EDS line is complete

No description available for this register.

TV_VBI_20BIT_CNTL - RW - 32 bits - [MMReg:0x8A0]			
Field Name	**Bits**	**Default**	**Description**
VBI_20BIT_DATA0	15:0	0x0	VBI Data for 20-bit Identification Stream. Note: Location of 20-bit ID line is determined by timing codes.
VBI_20BIT_DATA1	19:16	0x0	See above
VBI_20BIT_WT	24	0x0	Initiates a write cycle requested to 20-bit ID line using VBI_20BIT_DATA 0=write cycle inactive/complete (only complete when VBI_20BIT_WT_ACK is '1') 1=write cycle requested
VBI_20BIT_WT_ACK (R)	25	0x0	Indicates write cycle to 20-bit ID stream is complete 0=inactive (when no write request) or waiting for write cycle to complete (when write request initiated by VBI_20BIT_WT). 1=write cycle complete
VBI_20BIT_HOLD	26	0x0	Hold 20BIT data to value of VBI_20BIT_DATA 0=normal; each write will produce 20BIT data only once and if no data is written before the 20BIT line, the data will be null 1=every 20BIT line will use VBI_20BIT_DATA; VBI_20BIT_DATA is loaded rather than null when the 20BIT line is complete

No description available for this register.

TV_VBI_DTO_CNTL - RW - 32 bits - [MMReg:0x8A4]			
Field Name	**Bits**	**Default**	**Description**
VBI_CC_DTO_P	15:0	0x1802	DTO P value for Closed Caption (CC) and Extended Data Service (EDS/XDS) lines. This is an unsigned 4.12 number representing the step desired.P=128*(1/(desired period/ TV_CLK period))(e.g. NTSC: P=128*(1/(1986 ns/23.2804 ns))=1.5004...=>0x1802)
VBI_20BIT_DTO_P	31:16	0x155c	DTO P value for 20-bit ID line. This is an unsigned 4.12 number representing the step desired.P=128*(1/(desired period/TV_CLK period))

No description available for this register.

TV_VBI_LEVEL_CNTL - RW - 32 bits - [MMReg:0x8A8]			
Field Name	**Bits**	**Default**	**Description**
VBI_CC_LEVEL	6:0	0x3f	VBI CC/EDS level for ‘1’ data, unsigned 7-bit register
VBI_20BIT_LEVEL	14:8	0x59	VBI 20-bit ID level for ‘1’ data, unsigned 7-bit register
VBI_CLK_RUNIN_GAIN	24:16	0x10f	specifies gain to apply to CC sinusoidal clock run-in; default is to compensate for hardcoded ROM giving 46IRE whereIRE is desired

specifies digital levels for CC and CGMS signalling

TV_UV_ADR - RW - 32 bits - [MMReg:0x8AC]			
Field Name	**Bits**	**Default**	**Description**
MAX_UV_ADR	7:0	0xa0	Maximum address for UV data in fiforam. Pointer to Timing Code Table 1.
TABLE1_BOT_ADR	15:8	0x0	Pointer to Timing Code Table 2.
TABLE3_TOP_ADR	23:16	0x0	Pointer to Timing Code Table 3.
reserved	24		reserved
HCODE_TABLE_SEL	26:25	0x0	Horizontal Code Table Select. 0=Table 1 1=Table 2 2=Table 3 3=reserved
VCODE_TABLE_SEL	28:27	0x0	Vertical Code Table Select. 0=Table 1 1=Table 2 2=Table 3 3=reserved
reserved	30:29		reserved
SWITCH_TABLE_REQ	31	0x0	Request to accept new timing code tables, MV_DECODE_EN and MV_LINES PER_STRIPE selections on all following end of frame events 0=no request 1=request to accept

No description available for this register.

TV_VSYNC_DIFF_CNTL - RW - 32 bits - [MMReg:0x8F4]			
Field Name	**Bits**	**Default**	**Description**
VSYNC_DIFF_MEASURE_EN	0	0x0	Enable measurement of vsync difference. If 0, the TV timing will be restarted every frame (or every 2-8 fields for modes with 2-, 4- or 8-fields between repetitions of the timing sequence). 0=Don't measure 1=Measure difference in start of frames
VSYNC_RESTART_AT_LIMIT	1	0x0	Enable suppression of the forced restart of TV timing unless VSYNC_DIFF exceeds DIFF_LIMIT_HIGH. 0=Restart at every crt vsync if tv first, or every other if not 1=Restart when vsync difference greater than high limit
VSYNC_HW_LOCK_EN	2	0x0	Enable hardware feedback loop to lock the longterm frame rate of the TV output to that of the CRT. 0=Disable 1=Enable hardware feedback
VSYNC_HW_LOCK_TYPE	3	0x0	Select the mechanism to be used for hardware feedback loop. 0=Clutch Slip 1=M/N Flip
VSYNC_SLIP_REQ_EN	4	0x0	Select whether hardware slip requests are generated by a DTO or based on horizontal sync edges. 0=Use hardware DTO to generate slip requests 1=Use horizontal sync falling edge to trigger slips
INVERT_LOCK_CNTL	5	0x0	Allows inversion of the control signal for hardware feedback, in case the hardware was set up incorrectly. 0=Don't invert 1=Invert lock control signal
SLIP_DURING_HSYNC_ONLY	6	0x0	When hardware feedback loop is active, this bit prevents clutch slip requests from happening outside of the TV horizontal sync. 0=Slip any time 1=Slip during hsync only
RESTART_TWICE	7	0x0	If a restart is required because VSYNC_DIFF exceeds DIFF_LIMIT_HIGH, and the CRT vsync happened first, there are two possible locations to force the TV timing to restart. A restart can be made to happen immediately, but this will not set the vsync differenc to zero due to the time spent measuring the difference. The restart can also be delayed until the next CRT frame starts. If RESTART_TWICE is set to 1, the TV timing will be restarted at both of the possible locations; if RESTART_TWICE is 0, there will be only one restart signal sent, and its location will be selected by DELAY_RESTART. 0=Restart once 1=Restart twice if CRT first

DELAY_RESTART	8	0x0	(See above) this bit only has an effect if RESTART_TWICE = 0. When the CRT vsync happens first, this bit is used to determine whether to restart the TV timing as soon as VSYNC_DIFF has exceed DIFF_LIMIT_HIGH, or whether to wait until the next CRT vsync to send the restart signal. 0=Restart immediately 1=Restart on next CRT vsync

Controls circuits used to measure and correct the difference in the vertical sync times for the CRT and TV portions of TV out

TV_VSYNC_DIFF_LIMITS - RW - 32 bits - [MMReg:0x8F8]			
Field Name	**Bits**	**Default**	**Description**
DIFF_LIMIT_LOW	11:0	0x0	This field is only used if VSYNC_DIFF_MEASURE_EN = 1, VSYNC_RESTART_AT_LIMIT = 1, and VSYNC_HW_LOCK_EN = 1. If VSYNC_DIFF >= DIFF_LIMIT_LOW, the TV frame rate will be set slightly faster or slower than the CRT as required to correct the vsync difference. Otherwise no action is taken by the hardware feedback mechanism.
DIFF_LIMIT_HIGH	27:16	0x0	This field is only used if VSYNC_DIFF_MEASURE_EN = 1 and VSYNC_RESTART_AT_LIMIT = 1. If VSYNC_DIFF >= DIFF_LIMIT_HIGH, then a restart signal will be sent to the TV timing generator.

Constants to be used to compare with VSYNC_DIFF in order to determine what, if any, action needs to be taken in response to the measured difference.

TV_VSYNC_DIFF_RD_DATA - R - 32 bits - [MMReg:0x8FC]			
Field Name	**Bits**	**Default**	**Description**
UPDATE_END_TOGGLE	0	0x0	This field toggles after the fields VSYNC_DIFF, VSYNC_DIFF_OVERFLOW and TV_FIRST have been updated. Software should only read in order, first the lower byte(bits 7:0), then the upper byte(bits 15:8) and then check that UPDATE_END_TOGGLE = UPDATE_START_TOGGLE every time this register is read. If this condition is false, then the register was being updated at the time it was read. Therefore, the values read are invalid and the register read should be repeated.
VSYNC_DIFF	12:1	0x0	This is the difference between the last CRT and TV vertical sync times, measured in TV clock periods.

VSYNC_DIFF_OVERFLOW	13	0x0	If the VSYNC_DIFF counter overflows, this bit is set and remains set until the next time the vertical sync difference is measured. 0=No overflow 1=Vsync difference counter has overflowed
TV_FIRST	14	0x0	If 1, the TV vertical sync happened first, meaning the TV frame was shorter and the TV is faster, otherwise the CRT was first. 0=CRT frame started first, so the CRT is faster than the TV 1=TV frame started first, so the TV is faster than the CRT
UPDATE_START_TOGGLE	15	0x0	This bit toggles before the fields VSYNC_DIFF, VSYNC_DIFF_OVERFLOW and TV_FIRST are updated. Software should only read in order, first the lower byte(bits 7:0), then the upper byte(bits 15:8) and then check that UPDATE_START_TOGGLE = UPDATE_END_TOGGLE every time this register is read. If this condition is false, then the register was being updated at the time it was read. Therefore, the values read are invalid and the register read should be repeated.

Sign and magnitude of the difference between the vertical sync times, really the 'end of frame points', of the CRT and TV portions of TV out.

GEN_INT_CNTL - RW - 32 bits - [IOReg,MMReg:0x40]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_MASK (DISPLAY)	0	0x0	Vertical blank interrupt mask. 0=Disable 1=Enable
CRTC_VLINE_MASK (DISPLAY)	1	0x0	Vertical line interrupt mask. 0=Disable 1=Enable
CRTC_VSYNC_MASK (DISPLAY)	2	0x0	Vertical sync interrupt mask. 0=Disable 1=Enable
SNAPSHOT_MASK (DISPLAY)	3	0x0	Snapshot interrupt mask. 0=Disable 1=Enable
FP_DETECT_MASK (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt mask. 0=Disable 1=Enable
CRTC2_VLINE_MASK (DISPLAY)	5	0x0	0=Disable 1=Enable

CRTC2_VSYNC_MASK (DISPLAY)	6	0x0	0=Disable 1=Enable
SNAPSHOT2_MASK (DISPLAY)	7	0x0	0=Disable 1=Enable
CRTC2_VBLANK_MASK (DISPLAY)	9	0x0	0=Disable 1=Enable
FP2_DETECT_MASK (DISPLAY)	10	0x0	0=Disable 1=Enable
VSYNC_DIFF_OVER_LIMIT_MASK (TVOUT)	11	0x0	0=Disable 1=Enable
GUI_IDLE_MASK (RBBM)	19	0x0	GUI idle interrupt mask. 0=Disable 1=Enable
SW_INT_EN (HDP)	25	0x0	Software interrupt mask. 0=Disable 1=Enable
GEYSERVILLE_MASK (VIP)	27	0x0	0=Disable 1=Enable
HDCP_AUTHORIZED_INT_MASK (DISPLAY)	28	0x0	0=Disable 1=Enable
DVI_I2C_INT_MASK (DISPLAY)	29	0x0	0=Disable 1=Enable
GUIDMA_MASK (CP)	30	0x0	GUI DMA channel interrupt mask. 0=Disable 1=Enable
VIDDMA_MASK (CP)	31	0x0	Video capture DMA channel interrupt mask. 0=Disable 1=Enable

General Interrupt Control register.
The MASK/EN fields control whether the respective status bits are enabled to drive the system interrupt pin. Even if enabled here, the interrupt line is not driven unless enabled in the PCI configuration space.

GEN_INT_STATUS - RW - 32 bits - [IOReg,MMReg:0x44]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_STAT (R) (DISPLAY)	0	0x0	Vertical blank interrupt. Set when display in vertical retrace. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VBLANK_STAT_AK (W) (DISPLAY)	0	0x0	Vertical blank interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VLINE_STAT (R) (DISPLAY)	1	0x0	Vertical line interrupt. Set on display line on programmed by the CRTC_VLINE_CRNT_VLINE.CRTC_VLINE register. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VLINE_STAT_AK (W) (DISPLAY)	1	0x0	Vertical line interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VSYNC_STAT (R) (DISPLAY)	2	0x0	Vertical sync interrupt. Set on start of VSYNC at the DAC. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_STAT_AK (W) (DISPLAY)	2	0x0	Vertical sync interrupt acknowledge. 0=No effect 1=Clear status
SNAPSHOT_STAT (R) (DISPLAY)	3	0x0	Snapshot interrupt. Set as controlled by SNAPSHOT_VIF_COUNT register. 0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT_STAT_AK (W) (DISPLAY)	3	0x0	Snapshot interrupt acknowledge. 0=No effect 1=Clear status
FP_DETECT_STAT (R) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt. Set on HPD connect or disconnect as controlled by FP_GEN_CNTL.FP_DETECT_INT_POL. 0=No event 1=Event has occurred, interrupting if enabled
FP_DETECT_STAT_AK (W) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt acknowledge. 0=No effect 1=Clear status
CRTC2_VLINE_STAT (R) (DISPLAY)	5	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VLINE_STAT_AK (W) (DISPLAY)	5	0x0	0=No effect 1=Clear status

CRTC2_VSYNC_STAT (R) (DISPLAY)	6	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VSYNC_STAT_AK (W) (DISPLAY)	6	0x0	0=No effect 1=Clear status
SNAPSHOT2_STAT (R) (DISPLAY)	7	0x0	0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT2_STAT_AK (W) (DISPLAY)	7	0x0	0=No effect 1=Clear status
CAP0_INT_ACTIVE (R) (VIP)	8	0x0	Capture port 0 has active interrupt(s). 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CRTC2_VBLANK_STAT (R) (DISPLAY)	9	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VBLANK_STAT_AK (W) (DISPLAY)	9	0x0	0=No effect 1=Clear status
FP2_DETECT_STAT (R) (DISPLAY)	10	0x0	0=No event 1=Event has occurred, interrupting if enabled
FP2_DETECT_STAT_AK (W) (DISPLAY)	10	0x0	0=No effect 1=Clear status
VSYNC_DIFF_OVER_LIMIT_STAT (R) (TVOUT)	11	0x0	0=No event 1=Event has occurred, interrupting if enabled
VSYNC_DIFF_OVER_LIMIT_STAT_AK (W) (TVOUT)	11	0x0	0=No effect 1=Clear status
GUI_IDLE_STAT (R) (RBBM)	19	0x1	GUI idle interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_STAT_AK (W) (RBBM)	19	0x0	GUI idle interrupt acknowledge. 0=No effect 1=Clear status
SW_INT (R) (HDP)	25	0x0	Software interrupt. General purpose interrupt that can only be set by software event by writing to SW_INT_SET. 0=No event 1=Event has occurred, interrupting if enabled

SW_INT_AK (W) (HDP)	25	0x0	Software interrupt acknowledge. 0=No effect 1=Clear SW_INT (set low)
SW_INT_SET (W) (HDP)	26	0x0	Software interrupt trigger. 0=No effect 1=Set SW_INT active (high)
GEYSERVILLE_STAT (R) (VIP)	27	0x0	0=No event 1=Event has occurred, interrupting if enabled
GEYSERVILLE_STAT_AK (W) (VIP)	27	0x0	0=No effect 1=Clear status
HDCP_AUTHORIZED_INT_STAT (R) (DISPLAY)	28	0x0	0=No event 1=Event has occurred, interrupting if enabled
HDCP_AUTHORIZED_INT_AK (W) (DISPLAY)	28	0x0	0=No effect 1=Clear status
DVI_I2C_INT_STAT (R) (DISPLAY)	29	0x0	0=No event 1=Event has occurred, interrupting if enabled
DVI_I2C_INT_AK (W) (DISPLAY)	29	0x0	0=No effect 1=Clear status
GUIDMA_STAT (R) (CP)	30	0x0	GUI DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUIDMA_AK (W) (CP)	30	0x0	GUI DMA channel interrupt acknowledge. 0=No effect 1=Clear status
VIDDMA_STAT (R) (CP)	31	0x0	Video capture DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
VIDDMA_AK (W) (CP)	31	0x0	Video capture DMA channel interrupt acknowledge. 0=No effect 1=Clear status

General Interrupt Status register.
These fields can be polled and acknowledged even if interrupts are disabled, or the respective fields are masked in the GEN_INT_CNTL register.

2.40 HDPcontrol Registers

HOST_PATH_CNTL - RW - 32 bits - [MMReg:0x130]			
Field Name	**Bits**	**Default**	**Description**
HDP_APER_CNTL	23	0x0	Selects how the two PCI linear memory apertures map into the internal 32bit address space of the graphics controller. 0=Both host apertures map to same area in MC address space (starting at MC_FB_START). Surface ranges are limited to the size of CONFIG_APER_SIZE. 1=Two host apertures are mapped one above the other in MC address space stating at MC_FP_START. i.e. they become one big aperture. Surface ranges can cover the double size of CONFIG_APER_SIZE.
HP_LIN_RD_CACHE_DIS	24	0x0	Selects whether to disable HDP prefetching. 0=Linear aperture reads taken from HostDataPath cache, if possible. 1=Linear aperture reads always sent to memory.
HP_RBBM_LOCK_DIS	25	0x0	Controls coherency between linear aperture cycles and register cycles that affect linear aperture operation (e.g. changing SURFACE registers or VGA write/read mode)./nThis field should normally be 0 to ensure proper coherency, and should be changed only if needed to avoid deadlock problems. 0=The RBBM write requests will be held untile the data pipe is idle. 1=The RBBM write requests will not be held.
HDP_SOFT_RESET	26	0x0	Software reset for HDP. 0=HDP running in normal operating mode 1=Soft Reset to HDP
HDP_WRITE_COMBINER_TIMEOUT	30:28	0x0	Controls internal timeout for write combining optimization. This allows separate dword writes to be combined into octawords before sending to the internal memory controller. This reduces bandwidth cost of slave writes and increases performance./nIt is recommended this register be set to 0x7 (28 BCLKs) unless otherwise noted. 0=0 BCLK 1=4 BCLK 2=8 BCLK 3=12 BCLK 4=16 BCLK 5=20 BCLK 6=24 BCLK 7=28 BCLK
HP_TEST_RST_CNTL	31	0x0	For HW test and debugging only. No use to software.

Controls for the Merged Register Bus internal to the controller for non-FIFOed register writes, and all register reads

MEM_VGA_WP_SEL - RW - 32 bits - [IOReg,MMReg:0x38]			
Field Name	**Bits**	**Default**	**Description**
MEM_VGA_WPS0	11:0	0x0	Write page pointer for lower 32 KByte aperture into 128 MByte video memory.
MEM_VGA_WPS1	27:16	0x0	Write page pointer for upper 32 KByte aperture into 128 MByte video memory.

Two write page pointers used for the two small 32K apertures at 0xA000 and 0xA800. Pages are selectable only on 32K boundaries.

MEM_VGA_RP_SEL - RW - 32 bits - [IOReg,MMReg:0x3C]			
Field Name	**Bits**	**Default**	**Description**
MEM_VGA_RPS0	11:0	0x0	Read page pointer for lower 32 KByte aperture into 128 MByte video memory.
MEM_VGA_RPS1	27:16	0x0	Read page pointer for upper 32 KByte aperture into 128 MByte video memory.

Two read page pointers used for the two small 32K apertures at 0xA000 and 0xA800. Pages are selectable only on 32K boundaries.

SW_SEMAPHORE - RW - 32 bits - [MMReg:0x13C]			
Field Name	**Bits**	**Default**	**Description**
SW_SEMAPHORE	15:0	0x0	Scratch register for use by software to implement status flags and semaphores. No affect on the hardware.

Scratch register

HDP_DEBUG - RW - 32 bits - [MMReg:0x138]			
Field Name	**Bits**	**Default**	**Description**
HDP_0_DEBUG	0	0x0	Spare debug registers for future implementation as needed.
HDP_1_DEBUG	1	0x0	Spare debug registers for future implementation as needed.
HDP_2_DEBUG	2	0x0	Spare debug registers for future implementation as needed.
HDP_3_DEBUG	3	0x0	Spare debug registers for future implementation as needed.
HDP_4_DEBUG	4	0x0	Spare debug registers for future implementation as needed.

HDP_5_DEBUG	5	0x0	Spare debug registers for future implementation as needed.
HDP_6_DEBUG	6	0x0	Spare debug registers for future implementation as needed.
HDP_7_DEBUG	7	0x0	Spare debug registers for future implementation as needed.

Hardware debug register

GEN_INT_CNTL - RW - 32 bits - [IOReg,MMReg:0x40]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_MASK (DISPLAY)	0	0x0	Vertical blank interrupt mask. 0=Disable 1=Enable
CRTC_VLINE_MASK (DISPLAY)	1	0x0	Vertical line interrupt mask. 0=Disable 1=Enable
CRTC_VSYNC_MASK (DISPLAY)	2	0x0	Vertical sync interrupt mask. 0=Disable 1=Enable
SNAPSHOT_MASK (DISPLAY)	3	0x0	Snapshot interrupt mask. 0=Disable 1=Enable
FP_DETECT_MASK (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt mask. 0=Disable 1=Enable
CRTC2_VLINE_MASK (DISPLAY)	5	0x0	0=Disable 1=Enable
CRTC2_VSYNC_MASK (DISPLAY)	6	0x0	0=Disable 1=Enable
SNAPSHOT2_MASK (DISPLAY)	7	0x0	0=Disable 1=Enable
CRTC2_VBLANK_MASK (DISPLAY)	9	0x0	0=Disable 1=Enable
FP2_DETECT_MASK (DISPLAY)	10	0x0	0=Disable 1=Enable
VSYNC_DIFF_OVER_LIMIT_MASK (TVOUT)	11	0x0	0=Disable 1=Enable

GUI_IDLE_MASK (RBBM)	19	0x0	GUI idle interrupt mask. 0=Disable 1=Enable
SW_INT_EN (HDP)	25	0x0	Software interrupt mask. 0=Disable 1=Enable
GEYSERVILLE_MASK (VIP)	27	0x0	0=Disable 1=Enable
HDCP_AUTHORIZED_INT_MASK (DISPLAY)	28	0x0	0=Disable 1=Enable
DVI_I2C_INT_MASK (DISPLAY)	29	0x0	0=Disable 1=Enable
GUIDMA_MASK (CP)	30	0x0	GUI DMA channel interrupt mask. 0=Disable 1=Enable
VIDDMA_MASK (CP)	31	0x0	Video capture DMA channel interrupt mask. 0=Disable 1=Enable

General Interrupt Control register.
The MASK/EN fields control whether the respective status bits are enabled to drive the system interrupt pin. Even if enabled here, the interrupt line is not driven unless enabled in the PCI configuration space.

GEN_INT_STATUS - RW - 32 bits - [IOReg,MMReg:0x44]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_STAT (R) (DISPLAY)	0	0x0	Vertical blank interrupt. Set when display in vertical retrace. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VBLANK_STAT_AK (W) (DISPLAY)	0	0x0	Vertical blank interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VLINE_STAT (R) (DISPLAY)	1	0x0	Vertical line interrupt. Set on display line on programmed by the CRTC_VLINE_CRNT_VLINE.CRTC_VLINE register. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VLINE_STAT_AK (W) (DISPLAY)	1	0x0	Vertical line interrupt acknowledge. 0=No effect 1=Clear status

CRTC_VSYNC_STAT (R) (DISPLAY)	2	0x0	Vertical sync interrupt. Set on start of VSYNC at the DAC. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_STAT_AK (W) (DISPLAY)	2	0x0	Vertical sync interrupt acknowledge. 0=No effect 1=Clear status
SNAPSHOT_STAT (R) (DISPLAY)	3	0x0	Snapshot interrupt. Set as controlled by SNAPSHOT_VIF_COUNT register. 0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT_STAT_AK (W) (DISPLAY)	3	0x0	Snapshot interrupt acknowledge. 0=No effect 1=Clear status
FP_DETECT_STAT (R) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt. Set on HPD connect or disconnect as controlled by FP_GEN_CNTL.FP_DETECT_INT_POL. 0=No event 1=Event has occurred, interrupting if enabled
FP_DETECT_STAT_AK (W) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt acknowledge. 0=No effect 1=Clear status
CRTC2_VLINE_STAT (R) (DISPLAY)	5	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VLINE_STAT_AK (W) (DISPLAY)	5	0x0	0=No effect 1=Clear status
CRTC2_VSYNC_STAT (R) (DISPLAY)	6	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VSYNC_STAT_AK (W) (DISPLAY)	6	0x0	0=No effect 1=Clear status
SNAPSHOT2_STAT (R) (DISPLAY)	7	0x0	0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT2_STAT_AK (W) (DISPLAY)	7	0x0	0=No effect 1=Clear status
CAP0_INT_ACTIVE (R) (VIP)	8	0x0	Capture port 0 has active interrupt(s). 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CRTC2_VBLANK_STAT (R) (DISPLAY)	9	0x0	0=No event 1=Event has occurred, interrupting if enabled

CRTC2_VBLANK_STAT_AK (W) (DISPLAY)	9	0x0	0=No effect 1=Clear status
FP2_DETECT_STAT (R) (DISPLAY)	10	0x0	0=No event 1=Event has occurred, interrupting if enabled
FP2_DETECT_STAT_AK (W) (DISPLAY)	10	0x0	0=No effect 1=Clear status
VSYNC_DIFF_OVER_LIMIT_STAT (R) (TVOUT)	11	0x0	0=No event 1=Event has occurred, interrupting if enabled
VSYNC_DIFF_OVER_LIMIT_STAT_AK (W) (TVOUT)	11	0x0	0=No effect 1=Clear status
GUI_IDLE_STAT (R) (RBBM)	19	0x1	GUI idle interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_STAT_AK (W) (RBBM)	19	0x0	GUI idle interrupt acknowledge. 0=No effect 1=Clear status
SW_INT (R) (HDP)	25	0x0	Software interrupt. General purpose interrupt that can only be set by software event by writing to SW_INT_SET. 0=No event 1=Event has occurred, interrupting if enabled
SW_INT_AK (W) (HDP)	25	0x0	Software interrupt acknowledge. 0=No effect 1=Clear SW_INT (set low)
SW_INT_SET (W) (HDP)	26	0x0	Software interrupt trigger. 0=No effect 1=Set SW_INT active (high)
GEYSERVILLE_STAT (R) (VIP)	27	0x0	0=No event 1=Event has occurred, interrupting if enabled
GEYSERVILLE_STAT_AK (W) (VIP)	27	0x0	0=No effect 1=Clear status
HDCP_AUTHORIZED_INT_STAT (R) (DISPLAY)	28	0x0	0=No event 1=Event has occurred, interrupting if enabled
HDCP_AUTHORIZED_INT_AK (W) (DISPLAY)	28	0x0	0=No effect 1=Clear status

DVI_I2C_INT_STAT (R) (DISPLAY)	29	0x0	0=No event 1=Event has occurred, interrupting if enabled
DVI_I2C_INT_AK (W) (DISPLAY)	29	0x0	0=No effect 1=Clear status
GUIDMA_STAT (R) (CP)	30	0x0	GUI DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUIDMA_AK (W) (CP)	30	0x0	GUI DMA channel interrupt acknowledge. 0=No effect 1=Clear status
VIDDMA_STAT (R) (CP)	31	0x0	Video capture DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
VIDDMA_AK (W) (CP)	31	0x0	Video capture DMA channel interrupt acknowledge. 0=No effect 1=Clear status

General Interrupt Status register.
These fields can be polled and acknowledged even if interrupts are disabled, or the respective fields are masked in the GEN_INT_CNTL register.

CRTC_EXT_CNTL - RW - 32 bits - [IOReg,MMReg:0x54]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VGA_XOVERSCAN (DISPLAY)	0	0x0	Set low for VGA compatible borders. When set high extended overscan registers control border in VGA modes. See also auto-centering in CRTC_MORE_CNTL and FP_GEN_CNTL.CRTC_VGA_XOVERSCAN_COLOR. 0=Disable extended overscan in VGA 1=Enable extended overscan in VGA
VGA_BLINK_RATE (DISPLAY)	2:1	0x0	Controls number of frames per blink for VGA modes. 0=Default VGA blink rate (16 frames) 1=1/2 default VGA blink rate (32 frames) 2=1/3 default VGA blink rate (48 frames) 3=1/4 default VGA blink rate (64 frames)
VGA_ATI_LINEAR (HDP) (DISPLAY)	3	0x0	Enable linear addressing through VGA memory aperture. 0=Disable 1=Enable
VGA_128KAP_PAGING (HDP) (DISPLAY)	4	0x0	Enable extended aperture paging in 128K VGA aperture mode. 0=Normal 1=Enable

VGA_TEXT_132 (HDP) (DISPLAY)	5	0x0	Extended text mode select (linear address 132 column text mode). Set low for VGA compatible 40 or 80 column text modes. 0=inActive 1=Active
VGA_XCRT_CNT_EN (DISPLAY)	6	0x0	Extended CRTC display address counter enable. Active High 0=Disable 1=Enable Ext CRTC Counter
CRTC_HSYNC_DIS (DISPLAY)	8	0x0	Disables horizontal sync output. Could be used for DPMS signaling, but DISP_PWR_MAN.DISP_PWR_MAN_DPMS is recommended instead. 0=Enable 1=Disable
CRTC_VSYNC_DIS (DISPLAY)	9	0x0	Disables vertical sync output. Could be used for DPMS signaling, but DISP_PWR_MAN.DISP_PWR_MAN_DPMS is recommended instead. Note this must remain enabled while using composite SYNC on HSYNC (CRTC_C_SYNC_EN=1). 0=Enable 1=Disable
CRTC_DISPLAY_DIS (DISPLAY)	10	0x0	Disables the display, forcing the blanking signal to be active. When blanking the screen with this bit, the overlay, sub-picture, graphics and cursor should also be disabled to save power. See CRTC_DISP_REQ_EN_B, CRTC_CUR_EN, OV0_OVERLAY_EN and SUBPIC_ON. 0=Enable 1=Blanked
CRTC_SYNC_TRISTATE (DISPLAY)	11	0x0	Tristates HSYNC and VSYNC outputs. For individual tristate control, see the next two fields. 0=Normal 1=Tristate HSYNC and VSYNC outputs
CRTC_HSYNC_TRISTATE (DISPLAY)	12	0x0	Tristates HSYNC output. 0=Normal HSYNC 1=Tristate HSYNC output
CRTC_VSYNC_TRISTATE (DISPLAY)	13	0x0	Tristates VSYNC output. This is recommended for use with composite sync mode when only the HSYNC output pin should be enabled. 0=Normal VSYNC 1=Tristate VSYNC output
CRT_ON (DISPLAY)	15	0x0	0=CRT OFF 1=CRT ON
VGA_CUR_B_TEST (DISPLAY)	17	0x0	Test cursor blinking. Only used for diagnostic testing. 0=Disable VGA cursor test 1=Test VGA cursor blinking

VGA_PACK_DIS (HDP)	18	0x0	Controls host write pipe for packed VGA modes (e.g. mode 13). Should only be set high if HW problem with fast writes. 0=Fast VGA write in packed modes 1=Normal VGA write in packed modes
VGA_MEM_PS_EN (HDP)	19	0x0	VGA page select enable: 0=Don't use MEM_VGA_WP_SEL and MEM_VGA_RP_SEL registers 1=Use MEM_VGA_WP_SEL and MEM_VGA_RP_SEL registers
VCRTC_IDX_MASTER (HDP) (DISPLAY)	30:24	0x0	VGA CRTC master index. Only bits 5:0 of the VGA CRTC index can be written (or read) in VGA I/O space at 0x3B4 or 0x3D4. Bit 6 controls whether the master or shadow set of VGA CRTC registers is seen in VGA I/O space. The shadow set is for use when supporting panel operation in VGA modes. The BIOS will leave either the master or shadow set active as needed after a mode switch call.

More CRTC general controls

GENMO_RD - R - 8 bits - VGA_IO:0x3CC			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B *(mirror of GENMO_WT:GENMO_MONO_ADDRESS_B)* (BIF)	0	0x0	VGA addressing mode.
VGA_RAM_EN *(mirror of GENMO_WT:VGA_RAM_EN)* (BIF)	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture.
VGA_CKSEL *(mirror of GENMO_WT:VGA_CKSEL)* (DISPLAY)	3:2	0x0	Selects pixel clock frequency to use.
ODD_EVEN_MD_PGSEL *(mirror of GENMO_WT:ODD_EVEN_MD_PGSEL)* (HDP)	5	0x0	This bit is used in odd/even display modes (A/N modes: 0, 1, 2, 3, and 7). This bit is ignored when either bit GRA06[1] or SEQ4[3] are enabled. Used to determine if the VGA aperture maps into the lower (even) or upper (odd) page of memory.

VGA_HSYNC_POL *(mirror of GENMO_WT:VGA_HSYNC_POL)* (DISPLAY)	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low The covention of VGA is to use active low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL *(mirror of GENMO_WT:VGA_VSYNC_POL)* (DISPLAY)	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low The covention of VGA is to use active high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

Miscellaneous Output Register (Read)

GENMO_WT - W - 8 bits - VGA_IO:0x3C2			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B (BIF)	0	0x0	VGA addressing mode. 0=Monochrome emulation, regs at 0x3Bx 1=Color/Graphic emulation, regs at 0x3Dx
VGA_RAM_EN (BIF)	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture. 0=Disable 1=Enable
VGA_CKSEL (DISPLAY)	3:2	0x0	Selects pixel clock frequency to use in VGA modes. Used when CRTC_GEN_CNTL.CRTC_EXT_DISP_EN=0. See CLOCK_CNTL_INDEX.PPLL_DIV_SEL for non-VGA mode pixel clock selection. 0=25.1744MHz (640 Pels) 1=28.3212MHz (720 Pels) 2=Reserved 3=Reserved
ODD_EVEN_MD_PGSEL (HDP)	5	0x0	This bit is used in odd/even display modes (A/N modes: 0, 1, 2, 3, and 7). This bit is ignored when either bit GRA06[1] or SEQ4[3] are enabled. Used to determine if the VGA aperture maps into the lower (even) or upper (odd) page of memory. 0=Selects odd (high) memory locations 1=Selects even (low) memory locations

VGA_HSYNC_POL (DISPLAY)	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low The covention of VGA is to use active low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL (DISPLAY)	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low The covention of VGA is to use active high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

Miscellaneous Output Register (Write)

MC_FB_LOCATION - RW - 32 bits - [MMReg:0x148]			
Field Name	**Bits**	**Default**	**Description**
MC_FB_START (MC) (HDP)	15:0	0x0	Start of local frame buffer section of 32 bit internal address space. Recommend setting this register to same as CONFIG_APER_0_BASE.APER_0_BASE shr 16. NOTE: Bits 0:5 of this field are hardwired to ZERO.
MC_FB_TOP (MC)	31:16	0x3f	End of local frame buffer section of 32 bit internal address space. Recommand setting this register to (CONFIG_APER_0_BASE.APER_0_BASE + CONFIG_APER_SIZE.APER_SIZE - 1) shr 16. NOTE: Bits 0:5 of this field are hardwired to ONE.

This register defines the location of the frame buffer in the internal address space. The internal address space has 32 address bits. Minimum Frame buffer size for Rage 5 is 2 MB, and the start location is required to be on a 4 MB boundry. Therefore START(21:0) must be 0x000000 and TOP(21:0) must be 0x3FFFFF. Only the 16 MSBs of each are loaded in the register.

2.41 VGA_GRPH Registers

GRPH8_IDX - RW - 8 bits - VGA_IO:0x3CE			
Field Name	**Bits**	**Default**	**Description**
GRPH_IDX (HDP) (DISPLAY)	3:0	0x0	This index is used to address one of the internal registers of the graphics controller (GRAC) at I/O port 0x3CRF.

GRPH Index Register

GRPH8_DATA - RW - 8 bits - VGA_IO:0x3CF			
Field Name	**Bits**	**Default**	**Description**
GRPH_DATA (HDP) (DISPLAY)	7:0	0x0	GRPH data indirect access

GRPH Data Register

GRA00 - RW - 8 bits - VGAGRPHIND:0x0			
Field Name	**Bits**	**Default**	**Description**
GRPH_SET_RESET0	0	0x0	Set/Reset Map 0
GRPH_SET_RESET1	1	0x0	Set/Reset Map 1
GRPH_SET_RESET2	2	0x0	Set/Reset Map 2
GRPH_SET_RESET3	3	0x0	Set/Reset Map 3

Set/Reset Register

GRA01 - RW - 8 bits - VGAGRPHIND:0x1			
Field Name	**Bits**	**Default**	**Description**
GRPH_SET_RESET_ENA0	0	0x0	Enable Set/Reset Map 0
GRPH_SET_RESET_ENA1	1	0x0	Enable Set/Reset Map 1
GRPH_SET_RESET_ENA2	2	0x0	Enable Set/Reset Map 2
GRPH_SET_RESET_ENA3	3	0x0	Enable Set/Reset Map 3

Enable Set/Reset Register

GRA02 - RW - 8 bits - VGAGRPHIND:0x2			
Field Name	**Bits**	**Default**	**Description**
GRPH_CCOMP	3:0	0x0	Colour Compare Map bits 3:0. In Read mode (GRA05[3] being logical 1), the 4 bits from this register are compared with the 4-bit PEL value (made up of one bit from each map), from bit positions 0 through 7. As long as the colour don't care bits (GRA07[0:3]) for the respective maps are logical 1's, the compare takes place only on those bits of the PEL value, and the CPU reads a one for a match in that bit position. If Colour Don't Care bit for one map is a logical zero, the latched data from the map is excluded from the compare, and only the remaining three bits are compared to generate bus data.

Colour Compare Register

GRA03 - RW - 8 bits - VGAGRPHIND:0x3			
Field Name	**Bits**	**Default**	**Description**
GRPH_ROTATE	2:0	0x0	Rotate Count Bits 2-0. Specifies the number of bit positions that the CPU data is to be rotated to the right, before doing the function selected by bits 3 and 4 above and subsequency bit mask select and write operations. Rotation is carried out only in write modes 0 and 3. In these two modes, the CPU data is rotated first, the operated only the function bits GRA03[4:3], the updated by the bit mask register GRA05.
GRPH_FN_SEL	4:3	0x0	Function Select Bits 1 and 2. These funcitons are performed on the CPU data before the selected bits are updated by the bit mask register, and then written to the display buffers. 0=Replace 1=AND 2=OR 3=XOR

Data Rotate Register

GRA04 - RW - 8 bits - VGAGRPHIND:0x4			
Field Name	**Bits**	**Default**	**Description**
GRPH_RMAP	1:0	0x0	Read Mode 0 Only: GRA controller returns the contents of one of the four latched buffer bytes to CPU each time a CPU read loads these latches. The 2 bits (0 and 1) define a value that represents the bit map where CPU is to read data - usefull in transferring bit map data between the maps and system RAM.

Read Map Select Register

GRA05 - RW - 8 bits - VGAGRPHIND:0x5			
Field Name	**Bits**	**Default**	**Description**
GRPH_WRITE_MODE (HDP)	1:0	0x0	Write Mode: 0=Write mode 0 1=Write mode 1 2=Write mode 2 3=Write mode 3
GRPH_READ1 (HDP)	3	0x0	Read Mode: 0=Read mode 0, byte oriented 1=Read mode 1, pixel oriented
CGA_ODDEVEN (HDP)	4	0x0	Odd/Even Addressing Enable. Used to enable CGA emulation, this bit enables off/even addressing mode when it is logical one. Normally, this bit and memory mode bit SEQ04[2] are set to agree with each other in enabling odd/even mode emulation. 0=Disable Odd/Even Addressing 1=Enable Odd/Even Addressing
GRPH_OES (HDP) (DISPLAY)	5	0x0	Shift Register Mode: This bit controls how data from memory is loaded into the shift registers M0D0:M0D7, M1D0:M1D7; M2D0:M2D7, and M3D0:M3D7 are representations of this data. 0=Linear shift mode 1=Tiled shift mode
GRPH_PACK (HDP) (DISPLAY)	6	0x0	256 Colour Mode. This bit also contros how data from memory is loaded into the shift registers. 0=Use shift register mode as per GRPH_OES 1=256 color mode, read as packed pixels, ignore GRPH_OES

Graphics Mode Register

GRA06 - RW - 8 bits - VGAGRPHIND:0x6			
Field Name	**Bits**	**Default**	**Description**
GRPH_GRAPHICS	0	0x0	Graphics/Alphanumeric Mode 0=Alpha Numeric Mode 1=Graphics Mode
GRPH_ODDEVEN	1	0x0	Chains Odd Maps to Even 0=Normal 1=Chain Odd maps to Even
GRPH_ADRSEL	3:2	0x0	Memory Map Read Bits 1 and 0, respectively. 0=A0000-128K 1=A0000-64K 2=B0000-32K 3=B8000-32K

Graphics Miscellaneous Register

GRA07 - RW - 8 bits - VGAGRPHIND:0x7			
Field Name	**Bits**	**Default**	**Description**
GRPH_XCARE0	0	0x0	Ignore Map 0 0=Ignore map 0 1=Use map 0 for read mode 1
GRPH_XCARE1	1	0x0	Ignore Map 1 0=Ignore map 1 1=Use map 1 for read mode 1
GRPH_XCARE2	2	0x0	Ignore Map 2 0=Ignore map 2 1=Use map 2 for read mode 1
GRPH_XCARE3	3	0x0	Ignore Map 3 0=Ignore map 3 1=Use map 3 for read mode 1

Colour Don't Care Register

GRA08 - RW - 8 bits - VGAGRPHIND:0x8			
Field Name	**Bits**	**Default**	**Description**
GRPH_BMSK	7:0	0x0	Bit Mask

Bit Mask Register

2.42 VGA_SEQ Registers

SEQ8_IDX - RW - 8 bits - VGA_IO:0x3C4			
Field Name	**Bits**	**Default**	**Description**
SEQ_IDX (HDP) (DISPLAY)	2:0	0x0	This index points to one of the sequencer registers (SEQ_) at I/O port address 0x3C5, for the next SEQ read/write operation.

SEQ Index Register

SEQ8_DATA - RW - 8 bits - VGA_IO:0x3C5			
Field Name	**Bits**	**Default**	**Description**
SEQ_DATA (HDP) (DISPLAY)	7:0	0x0	SEQ data indirect access

SEQ Data Register

SEQ02 - RW - 8 bits - VGASEQIND:0x2			
Field Name	**Bits**	**Default**	**Description**
SEQ_MAP0_EN	0	0x0	Enable map 0 0=Disable write to memory map 0 1=Enable write to memory map 0
SEQ_MAP1_EN	1	0x0	Enable map 1 0=Disable write to memory map 1 1=Enable write to memory map 1
SEQ_MAP2_EN	2	0x0	Enable map 2 0=Disable write to memory map 2 1=Enable write to memory map 2
SEQ_MAP3_EN	3	0x0	Enable map 3 0=Disable write to memory map 3 1=Enable write to memory map 3

Map Mask Register

SEQ04 - RW - 8 bits - VGASEQIND:0x4			
Field Name	**Bits**	**Default**	**Description**
SEQ_256K	1	0x0	Extended memory - 1 indicates 256 KB of video memory is present. It also enables the character map selection in SEQ03. 0=64KB memory present. Has no effect since 256KB always available 1=256KB memory present
SEQ_ODDEVEN	2	0x0	Odd/Even 0=Even CPU address (A0=0) accesses maps 0 and 2. Odd address accesses maps 1 and 3 1=Enables sequential access to maps for odd/even modes. SEQ02 (Map Mask) selects which maps are used
SEQ_CHAIN	3	0x0	Chain (when logical 1, it takes priority over off/even mode bits SEQ04[2] and GRA05[4]. Unlike odd/even mode, SEQ04[2] is th eonly bit used to enable chain mode (double odd/even). Chain does not affect CRTC access to video memory. Odd/even bit SEQ04[2] should be the opposite of GRA05[4]. 0=Enables sequential access to maps. SEQ02 (Map Mask) selects which maps are used 1=For 256 color modes. Map select by CPU address bits A1:A0

Memory Mode Register

2.43 VGA_CRT Registers

CRTC8_IDX - RW - 8 bits - [VGA_IO:0x3B4] [VGA_IO:0x3D4]			
Field Name	**Bits**	**Default**	**Description**
VCRTC_IDX *(mirror bits 0:5 of CRTC_EXT_CNTL:VCRTC_IDX_MASTER)* (HDP) (DISPLAY)	5:0	0x0	This index points to one of the internal registers of the CRT controller (CRTC) at address 0x3?5, for the next CRTC read/write operation.

CRT Index Register

CRTC8_DATA - RW - 8 bits - [VGA_IO:0x3B5] [VGA_IO:0x3D5]			
Field Name	**Bits**	**Default**	**Description**
VCRTC_DATA (HDP) (DISPLAY)	7:0	0x0	CRTC data indirect access

CTRC Data Register

CRT14 - RW - 8 bits - VGACRTIND:0x14			
Field Name	**Bits**	**Default**	**Description**
UNDRLN_LOC (DISPLAY)	4:0	0x0	H Row Scan Bits 4-0.- These bits define the horizontal scan row, from the top of the characterline, that should be used for underlining. The 5-bit value is equal to the actual number minus one.
ADDR_CNT_BY4 (DISPLAY)	5	0x0	Count-by-4: 0=Char. Clock 1=CountBy4
DOUBLE_WORD (HDP) (DISPLAY)	6	0x0	Double-Word Mode: 0=Disable 1=DoubleWordMdEna

Underline Location Register

CRT17 - RW - 8 bits - VGACRTIND:0x17			
Field Name	**Bits**	**Default**	**Description**
RA0_AS_A13B (DISPLAY)	0	0x0	Compatibility Mode:
RA1_AS_A14B (DISPLAY)	1	0x0	Select Row Scan Counter:
VCOUNT_BY2 (DISPLAY)	2	0x0	Vertical_by_2 NOTE: When bit 2 is logical one, other vertical register values shoudl be adjusted as well (CRT06, CRT10, CRT12, CRT15, and CRT18).
ADDR_CNT_BY2 (DISPLAY)	3	0x0	Count_by_2: ENGINEERING NOTE: Bit can be written and read, but has no effect.
WRAP_A15TOA0 (DISPLAY)	5	0x0	Address Wrap: ENGINEERING NOTE: Bit can be written and read, but has no effect.
BYTE_MODE (HDP) (DISPLAY)	6	0x0	Byte/Word Mode: 0=WordMode 1=ByteMode
CRTC_SYNC_EN (DISPLAY)	7	0x0	H/V Retrace Enable: 0=Disable HVSync 1=EnaHVSync

CRT Mode Register

CRT1E - R - 8 bits - VGACRTIND:0x1E			
Field Name	**Bits**	**Default**	**Description**
GRPH_DEC_RD1	1	0x0	This register is used to read back the graphics controller index decode.

Graphics Controller Index Decode Register

CRT1F - R - 8 bits - VGACRTIND:0x1F			
Field Name	**Bits**	**Default**	**Description**
GRPH_DEC_RD0	7:0	0x0	This register is used to read back the graphics controller index decode.

Graphics Controller Index Decode Register

CRT22 - R - 8 bits - VGACRTIND:0x22			
Field Name	**Bits**	**Default**	**Description**
GRPH_LATCH_DATA	7:0	0x0	This register is used to read the data in the Graphics Controller CPU data latches. The Graphics Controller Read Map Select register bits 0 and 1 determines which byte is read back.

RAM Data Latch Readback Register

CRT14_S - RW - 8 bits - VGACRTIND:0x54			
Field Name	**Bits**	**Default**	**Description**
UNDRLN_LOC_S (DISPLAY)	4:0	0x0	Shadow copy UNDRLN_N_LOC
ADDR_CNT_BY4_M *(mirror of CRT14:ADDR_CNT_BY4)* (DISPLAY)	5	0x0	Mirror of ADDR_CNT_BY4
DOUBLE_WORD_M *(mirror of CRT14:DOUBLE_WORD)* (HDP) (DISPLAY)	6	0x0	Mirror of DOUBLE_WORD

Shadow of Underline Location Register

CRT17_S - RW - 8 bits - VGACRTIND:0x57			
Field Name	**Bits**	**Default**	**Description**
RA0_AS_A13B_M *(mirror of CRT17:RA0_AS_A13B)* (DISPLAY)	0	0x0	Mirror of RA0_AS_A13B
RA1_AS_A14B_M *(mirror of CRT17:RA1_AS_A14B)* (DISPLAY)	1	0x0	Mirror of RA1_AS_A14B

VCOUNT_BY2_S (DISPLAY)	2	0x0	Shadow copy of VCOUNT_BY2
ADDR_CNT_BY2_M *(mirror of CRT17:ADDR_CNT_BY2)* (DISPLAY)	3	0x0	Mirror of ADDR_CNT_BY2
WRAP_A15TOA0_M *(mirror of CRT17:WRAP_A15TOA0)* (DISPLAY)	5	0x0	Mirror of WRAP_A15TOA0
BYTE_MODE_M *(mirror of CRT17:BYTE_MODE)* (HDP) (DISPLAY)	6	0x0	Mirror of BYTE_MODE
CRTC_SYNC_EN_M *(mirror of CRT17:CRTC_SYNC_EN)* (DISPLAY)	7	0x0	Mirror of CRTC_SYNC_EN

Shadow of CRT Mode Register

CRT1E_S - R - 8 bits - VGACRTIND:0x5E			
Field Name	**Bits**	**Default**	**Description**
GRPH_DEC_RD1_M *(mirror of CRT1E:GRPH_DEC_RD1)*	1	0x0	Mirror of GRPH_DEC_RD1

Shadow of CRT1E

CRT1F_S - R - 8 bits - VGACRTIND:0x5F			
Field Name	**Bits**	**Default**	**Description**
GRPH_DEC_RD0_M *(mirror of CRT1F:GRPH_DEC_RD0)*	7:0	0x0	Mirror of GRPH_DEC_RD0

Shadow of CRT1F

CRT22_S - R - 8 bits - VGACRTIND:0x62			
Field Name	**Bits**	**Default**	**Description**
GRPH_LATCH_DATA_M *(mirror of CRT22:GRPH_LATCH_DATA)*	7:0	0x0	Mirror of GRPH_LATCH_DATA

Shadow of CRT22

2.44 Tiling Registers

Frame Buffer Tiling Control Registers.

SURFACE_CNTL - RW - 32 bits - [MMReg:0xB00]			
Field Name	**Bits**	**Default**	**Description**
SURF_TRANSLATION_DIS	8	0x1	Disable register for Tiling address translation. 0=Enable tiling translation based on SURFACE registers. 1=Disable tilingi based on SURFACE registers. Swapping still based on SURFACE comparisons and settings.
NONSURF_AP0_SWP	21:20	0x0	Endian Swap control for Aperture 0 accesses not in a surface. Always used when SURF_TRANSLATION_DIS is set to 1. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
NONSURF_AP1_SWP	23:22	0x0	Endian Swap control for Aperture 1 accesses not in a surface. Always used when SURF_TRANSLATION_DIS is set to 1. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.

General Tiling Control Registers

SURFACE_ACCESS_FLAGS - R - 32 bits - [MMReg:0xBF8]			
Field Name	**Bits**	**Default**	**Description**
SURF0_WRITE_FLAG	0	0x0	Indicates if write to surface 0 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF1_WRITE_FLAG	1	0x0	Indicates if write to surface 1 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF2_WRITE_FLAG	2	0x0	Indicates if write to surface 2 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF3_WRITE_FLAG	3	0x0	Indicates if write to surface 3 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.

SURF4_WRITE_FLAG	4	0x0	Indicates if write to surface 4 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF5_WRITE_FLAG	5	0x0	Indicates if write to surface 5 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF6_WRITE_FLAG	6	0x0	Indicates if write to surface 6 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF7_WRITE_FLAG	7	0x0	Indicates if write to surface 7 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
NONSURF_WRITE_FLAG	8	0x0	Indicates if write to address that missed all surfaces since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
LINEAR_WRITE_FLAG	9	0x0	Indicates if write to anywhere in linear memory aperture since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
VGA_WRITE_FLAG	10	0x0	Indicates if write to anywhere in VGA memory aperture since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF0_READ_FLAG	16	0x0	Indicates if read to surface 0 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
SURF1_READ_FLAG	17	0x0	Indicates if read to surface 1 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
SURF2_READ_FLAG	18	0x0	Indicates if read to surface 2 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
SURF3_READ_FLAG	19	0x0	Indicates if read to surface 3 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
SURF4_READ_FLAG	20	0x0	Indicates if read to surface 4 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
SURF5_READ_FLAG	21	0x0	Indicates if read to surface 5 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
SURF6_READ_FLAG	22	0x0	Indicates if read to surface 6 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

SURF7_READ_FLAG	23	0x0	Indicates if read to surface 7 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
NONSURF_READ_FLAG	24	0x0	Indicates if read to address that missed all surfaces since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
LINEAR_READ_FLAG	25	0x0	Indicates if read to anywhere in linear memory aperture since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.
VGA_READ_FLAG	26	0x0	Indicates if read to anywhere in VGA memory aperture since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Flags to indicate write or read access to memory surfaces

SURFACE_ACCESS_CLR - W - 32 bits - [MMReg:0xBFC]			
Field Name	**Bits**	**Default**	**Description**
SURF0_WRITE_FLAG_CLR	0	0x0	Used to clear SURF0_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF1_WRITE_FLAG_CLR	1	0x0	Used to clear SURF1_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF2_WRITE_FLAG_CLR	2	0x0	Used to clear SURF2_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF3_WRITE_FLAG_CLR	3	0x0	Used to clear SURF3_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF4_WRITE_FLAG_CLR	4	0x0	Used to clear SURF4_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF5_WRITE_FLAG_CLR	5	0x0	Used to clear SURF5_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF6_WRITE_FLAG_CLR	6	0x0	Used to clear SURF6_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.

SURF7_WRITE_FLAG_CLR	7	0x0	Used to clear SURF7_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
NONSURF_WRITE_FLAG_CLR	8	0x0	Used to clear NONSURF_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
LINEAR_WRITE_FLAG_CLR	9	0x0	Used to clear LINEAR_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
VGA_WRITE_FLAG_CLR	10	0x0	Used to clear VGA_WRITE_FLAG. 0=Do not clear this write flag. 1=Clear this write flag.
SURF0_READ_FLAG_CLR	16	0x0	Used to clear SURF0_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF1_READ_FLAG_CLR	17	0x0	Used to clear SURF1_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF2_READ_FLAG_CLR	18	0x0	Used to clear SURF2_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF3_READ_FLAG_CLR	19	0x0	Used to clear SURF3_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF4_READ_FLAG_CLR	20	0x0	Used to clear SURF4_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF5_READ_FLAG_CLR	21	0x0	Used to clear SURF5_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF6_READ_FLAG_CLR	22	0x0	Used to clear SURF6_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
SURF7_READ_FLAG_CLR	23	0x0	Used to clear SURF7_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
NONSURF_READ_FLAG_CLR	24	0x0	Used to clear NONSURF_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
LINEAR_READ_FLAG_CLR	25	0x0	Used to clear LINEAR_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.
VGA_READ_FLAG_CLR	26	0x0	Used to clear VGA_READ_FLAG. 0=Do not clear this read flag. 1=Clear this read flag.

Clear bits for SURFACE_ACCESS_FLAGS register

SURFACE0_LOWER_BOUND - RW - 32 bits - [MMReg:0xB04]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 0. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 0 Lower Bound Register

SURFACE1_LOWER_BOUND - RW - 32 bits - [MMReg:0xB14]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 1. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 1 Lower Bound Register

SURFACE2_LOWER_BOUND - RW - 32 bits - [MMReg:0xB24]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 2. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 2 Lower Bound Register

SURFACE3_LOWER_BOUND - RW - 32 bits - [MMReg:0xB34]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 3. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 3 Lower Bound Register

SURFACE4_LOWER_BOUND - RW - 32 bits - [MMReg:0xB44]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 4. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 4 Lower Bound Register

SURFACE5_LOWER_BOUND - RW - 32 bits - [MMReg:0xB54]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 5. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 5 Lower Bound Register

SURFACE6_LOWER_BOUND - RW - 32 bits - [MMReg:0xB64]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 6. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 6 Lower Bound Register

SURFACE7_LOWER_BOUND - RW - 32 bits - [MMReg:0xB74]			
Field Name	**Bits**	**Default**	**Description**
SURF_LOWER	27:0	0x0	Starting address of lowest tile in Surface 7. NOTE: Bits 0:10 of this field are hardwired to ZERO.

Surface 7 Lower Bound Register

SURFACE0_UPPER_BOUND - RW - 32 bits - [MMReg:0xB08]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 0. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 0 Upper Bound Register

SURFACE1_UPPER_BOUND - RW - 32 bits - [MMReg:0xB18]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 1. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 1 Upper Bound Register

SURFACE2_UPPER_BOUND - RW - 32 bits - [MMReg:0xB28]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 2. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 2 Upper Bound Register

SURFACE3_UPPER_BOUND - RW - 32 bits - [MMReg:0xB38]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 3. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 3 Upper Bound Register

SURFACE4_UPPER_BOUND - RW - 32 bits - [MMReg:0xB48]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 4. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 4 Upper Bound Register

SURFACE5_UPPER_BOUND - RW - 32 bits - [MMReg:0xB58]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 5. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 5 Upper Bound Register

SURFACE6_UPPER_BOUND - RW - 32 bits - [MMReg:0xB68]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 6. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 6 Upper Bound Register

SURFACE7_UPPER_BOUND - RW - 32 bits - [MMReg:0xB78]			
Field Name	**Bits**	**Default**	**Description**
SURF_UPPER	27:0	0x7ff	Starting address of highest tile in Surface 7. NOTE: Bits 0:10 of this field are hardwired to ONE.

Surface 7 Upper Bound Register

SURFACE0_INFO - RW - 32 bits - [MMReg:0xB0C]			
Field Name	**Bits**	**Default**	**Description**
SURF0_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 0. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.
SURF0_TILE_MODE	17:16	0x0	Mode of tiling for Surface 0. Set SURF0_PITCHSEL to 0 to disable tiling surface 0. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF0_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 0. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF0_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 0. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF0_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF0_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 0 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF0_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF0_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 0 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 0

SURFACE1_INFO - RW - 32 bits - [MMReg:0xB1C]			
Field Name	**Bits**	**Default**	**Description**
SURF1_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 1. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.
SURF1_TILE_MODE	17:16	0x0	Mode of tiling for Surface 1. Set SURF1_PITCHSEL to 0 to disable tiling surface 1. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF1_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 1. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF1_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 1. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF1_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF1_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 1 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF1_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF1_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 1 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 1

SURFACE2_INFO - RW - 32 bits - [MMReg:0xB2C]			
Field Name	**Bits**	**Default**	**Description**
SURF2_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 2. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.

SURF2_TILE_MODE	17:16	0x0	Mode of tiling for Surface 2. Set SURF2_PITCHSEL to 0 to disable tiling surface 2. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF2_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 2. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF2_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 2. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF2_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF2_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 2 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF2_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF2_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 2 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 2

SURFACE3_INFO - RW - 32 bits - [MMReg:0xB3C]			
Field Name	**Bits**	**Default**	**Description**
SURF3_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 3. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.
SURF3_TILE_MODE	17:16	0x0	Mode of tiling for Surface 3. Set SURF3_PITCHSEL to 0 to disable tiling surface 3. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF3_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 3. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.

SURF3_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 3. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF3_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF3_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 3 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF3_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF3_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 3 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 3

SURFACE4_INFO - RW - 32 bits - [MMReg:0xB4C]			
Field Name	**Bits**	**Default**	**Description**
SURF4_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 4. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.
SURF4_TILE_MODE	17:16	0x0	Mode of tiling for Surface 4. Set SURF4_PITCHSEL to 0 to disable tiling surface 4. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF4_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 4. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF4_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 4. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF4_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF4_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 4 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.

SURF4_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF4_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 4 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 4

SURFACE5_INFO - RW - 32 bits - [MMReg:0xB5C]			
Field Name	**Bits**	**Default**	**Description**
SURF5_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 5. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.
SURF5_TILE_MODE	17:16	0x0	Mode of tiling for Surface 5. Set SURF5_PITCHSEL to 0 to disable tiling surface 5. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF5_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 5. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF5_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 5. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF5_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF5_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 5 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF5_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF5_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 5 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 5

SURFACE6_INFO - RW - 32 bits - [MMReg:0xB6C]			
Field Name	**Bits**	**Default**	**Description**
SURF6_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 6. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.
SURF6_TILE_MODE	17:16	0x0	Mode of tiling for Surface 6. Set SURF6_PITCHSEL to 0 to disable tiling surface 6. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF6_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 6. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF6_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 6. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF6_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF6_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 6 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF6_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF6_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 6 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 6

SURFACE7_INFO - RW - 32 bits - [MMReg:0xB7C]			
Field Name	**Bits**	**Default**	**Description**
SURF7_PITCHSEL	9:0	0x0	Pitch in octawords (16 bytes) of Surface 7. A value of 0 disables tiling in Surface 0./nFor color macro tiling and color macro&micro tiling the pitch must be in 16 octaword increments from 16 to 512./nFor 32-bit Z tiling the pitch must be in 2 octaword increments from 2 to 514./n For 16-bit Z tiling the pitch may be in 1 octoword increments from 1 to 257./nPitch bits less than min increment are ignored.

SURF7_TILE_MODE	17:16	0x0	Mode of tiling for Surface 7. Set SURF7_PITCHSEL to 0 to disable tiling surface 7. 0=Disable MicroTiling 1=Enable MicroTiling 2=32 bit Z tiling 3=16 bit Z tiling
SURF7_AP0_SWP	21:20	0x0	Endian swap control for Aperture 0 accesses in Surface 7. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF7_AP1_SWP	23:22	0x0	Endian swap control for Aperture 1 accesses in Surface 7. 0=Little endian: no swapping. 1=Big endian: 16-bit swapping. 2=Big endian: 32-bit swapping.
SURF7_WRITE_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF7_WRITE_FLAG)* (R)	24	0x0	Indicates if write to surface 7 since last cleared. 0=No write since last cleared. 1=Write occured since last cleared.
SURF7_READ_FLAG *(mirror of SURFACE_ACCESS_FLAGS:SURF7_READ_FLAG)* (R)	25	0x0	Indicates if read to surface 7 since last cleared. 0=No read since last cleared. 1=Read occured since last cleared.

Control register for Surface 7

2.45 Capture Registers

Registers to facilitate the capture of input video data

FCP_CNTL - RW - 32 bits - [MMReg:0x910]			
Field Name	**Bits**	**Default**	**Description**
FCP0_SRC_SEL	2:0	0x4	PCICLK,PCLK, PCLKb, HREF, GND, HREFb. 0=PCICLK 1=PCLK 2=PCLKb 3=HREF 4=GND 5=HREFb

Capture Port FCP clock mux control

GEN_INT_STATUS - RW - 32 bits - [IOReg,MMReg:0x44]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_STAT (R) (DISPLAY)	0	0x0	Vertical blank interrupt. Set when display in vertical retrace. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VBLANK_STAT_AK (W) (DISPLAY)	0	0x0	Vertical blank interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VLINE_STAT (R) (DISPLAY)	1	0x0	Vertical line interrupt. Set on display line on programmed by the CRTC_VLINE_CRNT_VLINE.CRTC_VLINE register. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VLINE_STAT_AK (W) (DISPLAY)	1	0x0	Vertical line interrupt acknowledge. 0=No effect 1=Clear status
CRTC_VSYNC_STAT (R) (DISPLAY)	2	0x0	Vertical sync interrupt. Set on start of VSYNC at the DAC. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_STAT_AK (W) (DISPLAY)	2	0x0	Vertical sync interrupt acknowledge. 0=No effect 1=Clear status
SNAPSHOT_STAT (R) (DISPLAY)	3	0x0	Snapshot interrupt. Set as controlled by SNAPSHOT_VIF_COUNT register. 0=No event 1=Event has occurred, interrupting if enabled

SNAPSHOT_STAT_AK (W) (DISPLAY)	3	0x0	Snapshot interrupt acknowledge. 0=No effect 1=Clear status
FP_DETECT_STAT (R) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt. Set on HPD connect or disconnect as controlled by FP_GEN_CNTL.FP_DETECT_INT_POL. 0=No event 1=Event has occurred, interrupting if enabled
FP_DETECT_STAT_AK (W) (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt acknowledge. 0=No effect 1=Clear status
CRTC2_VLINE_STAT (R) (DISPLAY)	5	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VLINE_STAT_AK (W) (DISPLAY)	5	0x0	0=No effect 1=Clear status
CRTC2_VSYNC_STAT (R) (DISPLAY)	6	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VSYNC_STAT_AK (W) (DISPLAY)	6	0x0	0=No effect 1=Clear status
SNAPSHOT2_STAT (R) (DISPLAY)	7	0x0	0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT2_STAT_AK (W) (DISPLAY)	7	0x0	0=No effect 1=Clear status
CAP0_INT_ACTIVE (R) (VIP)	8	0x0	Capture port 0 has active interrupt(s). 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CRTC2_VBLANK_STAT (R) (DISPLAY)	9	0x0	0=No event 1=Event has occurred, interrupting if enabled
CRTC2_VBLANK_STAT_AK (W) (DISPLAY)	9	0x0	0=No effect 1=Clear status
FP2_DETECT_STAT (R) (DISPLAY)	10	0x0	0=No event 1=Event has occurred, interrupting if enabled
FP2_DETECT_STAT_AK (W) (DISPLAY)	10	0x0	0=No effect 1=Clear status

VSYNC_DIFF_OVER_LIMIT_STAT (R) (TVOUT)	11	0x0	0=No event 1=Event has occurred, interrupting if enabled
VSYNC_DIFF_OVER_LIMIT_STAT_AK (W) (TVOUT)	11	0x0	0=No effect 1=Clear status
GUI_IDLE_STAT (R) (RBBM)	19	0x1	GUI idle interrupt. 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_STAT_AK (W) (RBBM)	19	0x0	GUI idle interrupt acknowledge. 0=No effect 1=Clear status
SW_INT (R) (HDP)	25	0x0	Software interrupt. General purpose interrupt that can only be set by software event by writing to SW_INT_SET. 0=No event 1=Event has occurred, interrupting if enabled
SW_INT_AK (W) (HDP)	25	0x0	Software interrupt acknowledge. 0=No effect 1=Clear SW_INT (set low)
SW_INT_SET (W) (HDP)	26	0x0	Software interrupt trigger. 0=No effect 1=Set SW_INT active (high)
GEYSERVILLE_STAT (R) (VIP)	27	0x0	0=No event 1=Event has occurred, interrupting if enabled
GEYSERVILLE_STAT_AK (W) (VIP)	27	0x0	0=No effect 1=Clear status
HDCP_AUTHORIZED_INT_STAT (R) (DISPLAY)	28	0x0	0=No event 1=Event has occurred, interrupting if enabled
HDCP_AUTHORIZED_INT_AK (W) (DISPLAY)	28	0x0	0=No effect 1=Clear status
DVI_I2C_INT_STAT (R) (DISPLAY)	29	0x0	0=No event 1=Event has occurred, interrupting if enabled
DVI_I2C_INT_AK (W) (DISPLAY)	29	0x0	0=No effect 1=Clear status
GUIDMA_STAT (R) (CP)	30	0x0	GUI DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled

GUIDMA_AK (W) (CP)	30	0x0	GUI DMA channel interrupt acknowledge. 0=No effect 1=Clear status
VIDDMA_STAT (R) (CP)	31	0x0	Video capture DMA channel interrupt. 0=No event 1=Event has occurred, interrupting if enabled
VIDDMA_AK (W) (CP)	31	0x0	Video capture DMA channel interrupt acknowledge. 0=No effect 1=Clear status

General Interrupt Status register.
These fields can be polled and acknowledged even if interrupts are disabled, or the respective fields are masked in the GEN_INT_CNTL register.

CAP0_BUF0_OFFSET - RW - 32 bits - [MMReg:0x920]			
Field Name	**Bits**	**Default**	**Description**
CAP_BUF0_OFFSET	31:0	0x0	Capture Port 0 Buffer 0 starting address NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture Port 0 Buffer 0 starting address

CAP0_BUF1_OFFSET - RW - 32 bits - [MMReg:0x924]			
Field Name	**Bits**	**Default**	**Description**
CAP_BUF1_OFFSET	31:0	0x0	Capture Port 0 Buffer 1 starting address NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture Port 0 Buffer 1 starting address

CAP0_BUF0_EVEN_OFFSET - RW - 32 bits - [MMReg:0x928]			
Field Name	**Bits**	**Default**	**Description**
CAP_BUF0_EVEN_OFFSET	31:0	0x0	Capture Port 0 Buffer 0 even frame starting address NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture Port 0 Buffer 0 even frame starting address

CAP0_BUF1_EVEN_OFFSET - RW - 32 bits - [MMReg:0x92C]			
Field Name	**Bits**	**Default**	**Description**
CAP_BUF1_EVEN_OFFSET	31:0	0x0	Capture Port 0 Buffer 1 even frame starting address NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture Port 0 Buffer 1 even frame starting address

CAP0_BUF_PITCH - RW - 32 bits - [MMReg:0x930]			
Field Name	**Bits**	**Default**	**Description**
CAP_BUF_PITCH	11:0	0x0	Capture 0 buffer's pitch. NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture 0 buffer's pitch.

CAP0_V_WINDOW - RW - 32 bits - [MMReg:0x934]			
Field Name	**Bits**	**Default**	**Description**
CAP_V_START	11:0	0x0	Vertical window starting line number.
CAP_V_END	27:16	0x0	Vertical window end line number.

Capture 0's Vertical window.

CAP0_H_WINDOW - RW - 32 bits - [MMReg:0x938]			
Field Name	**Bits**	**Default**	**Description**
CAP_H_START	11:0	0x0	Horizontal window's start.
CAP_H_WIDTH	27:16	0x0	Horizontal window's width. NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture 0's Horizontal window.

CAP0_VBI0_OFFSET - RW - 32 bits - [MMReg:0x93C]			
Field Name	**Bits**	**Default**	**Description**
CAP_VBI0_OFFSET	31:0	0x0	Capture 0 VBI 0 buffer's starting address. NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture 0 VBI 0 buffer's starting address.

CAP0_VBI1_OFFSET - RW - 32 bits - [MMReg:0x940]			
Field Name	**Bits**	**Default**	**Description**
CAP_VBI1_OFFSET	31:0	0x0	Capture 0 VBI 1 buffer's starting address. NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture 0 VBI 1 buffer's starting address.

CAP0_VBI_V_WINDOW - RW - 32 bits - [MMReg:0x944]			
Field Name	**Bits**	**Default**	**Description**
CAP_VBI_V_START	11:0	0x0	Capture 0 VBI's Vertical start.
CAP_VBI_V_END	27:16	0x0	Capture 0 VBI's Vertical End.

Capture 0 VBI's vertical window

CAP0_VBI_H_WINDOW - RW - 32 bits - [MMReg:0x948]			
Field Name	**Bits**	**Default**	**Description**
CAP_VBI_H_START	11:0	0x0	Capture 0 VBI's Horizontal start.
CAP_VBI_H_WIDTH	27:16	0x0	Capture 0 VBI's Horizontal Width. NOTE: Bits 0:1 of this field are hardwired to ZERO.

Capture 0 VBI's horizontal window

CAP0_PORT_MODE_CNTL - RW - 32 bits - [MMReg:0x94C]			
Field Name	**Bits**	**Default**	**Description**
CAP_PORT_WIDTH	1	0x0	Capture 0 port width. 0=8 bits 1=16 bits
CAP_PORT_BYTE_USED	2	0x0	In 8 bit width mode, which byte used. 0=lower byte used 1=upper byte used

Capture 0 mode control register.

CAP0_TRIG_CNTL - RW - 32 bits - [MMReg:0x950]			
Field Name	**Bits**	**Default**	**Description**
CAP_TRIGGER_R (R)	1:0	0x0	Read only. Capture status. 0=capture complete 1=capture pending 2=capture in progress
CAP_TRIGGER_W (W)	0	0x0	Write only. Start capture next frame. 0=no action 1=capture next field/frame
CAP_EN	4	0x0	Capture 0 enable. 0=disable 1=enable
CAP_VSYNC_CNT (R)	15:8	0x0	Read only. VSYNC counter.
CAP_VSYNC_CLR	16	0x0	Reset the VSYNC counter.

Capture 0 trigger control.

CAP0_DEBUG - RW - 32 bits - [MMReg:0x954]			
Field Name	**Bits**	**Default**	**Description**
CAP_H_STATUS (R)	11:0	0x0	Capture 0 Horizontal status.
CAP_V_STATUS (R)	27:16	0x0	Capture 0 vertical status.
CAP_V_SYNC (R)	28	0x0	Capture 0 VSYNC status.
CAP_BUF_OVERFLOW_STATUS (R)	29	0x0	0=No cursor overflow 1=Capture buffer has overflowed since last cleared
CAP_BUF_OVERFLOW_CLR (W)	30	0x0	0=Write 0, no effect 1=Write 1, clear capture buffer overflow flag

Capture 0 debug status register.

CAP0_CONFIG - RW - 32 bits - [MMReg:0x958]			
Field Name	**Bits**	**Default**	**Description**
CAP_INPUT_MODE	0	0x0	Input mode. 0=OneShot trigger mode 1=Enable continuous capture
CAP_START_FIELD	1	0x0	Starting field. 0=Odd 1=Even
CAP_START_BUF_R (R)	2	0x0	Read only. Current starting buffer. 0=Buffer 0 1=Buffer 1
CAP_START_BUF_W (W)	3	0x0	Write only. Control starting buffer. 0=Buffer 0 1=Buffer 1
CAP_BUF_TYPE	5:4	0x0	Buffer type. 0=Field 1=Alternating 2=Frame
CAP_ONESHOT_MODE	6	0x0	ONESHOT mode. 0=FIELD 1=FRAME
CAP_BUF_MODE	8:7	0x0	Capture 0 buffer mode. 0=Single 1=Double 2=Triple
CAP_MIRROR_EN	9	0x0	Capture 0 mirroring function enable. 0=Normal 1=Mirror
CAP_ONESHOT_MIRROR_EN	10	0x0	ONESHOT buffer mirroring function enable. 0=Normal 1=Mirror
CAP_VIDEO_SIGNED_UV	11	0x0	Enable conversion to signed value. 1=Convert to signed
CAP_VBI_EN	13	0x0	VBI enable. 0=disable 1=enable
CAP_SOFT_PULL_DOWN_EN	14	0x0	Software pull down enable. 0=disable 1=enable

CAP_VIP_EXTEND_FLAG_EN	15	0x0	Extended flag enable. 0=DISABLE 1=ENABLE
CAP_FAKE_FIELD_EN	16	0x1	Fake field enable. 0=DISABLE 1=ENABLE
CAP_FIELD_START_LINE_DIFF	18:17	0x0	Odd, Even frame line number differences. 0=EQUAL 1=ODD_ONE_MORE_LINE 2=EVEN_ONE_MORE_LINE
CAP_HORZ_DOWN	20:19	0x0	Horizontal decimation. 0=Normal 1=x2 2=x4
CAP_VERT_DOWN	22:21	0x0	Vertical decimation. 0=Normal 1=x2 2=x4
CAP_STREAM_FORMAT	25:23	0x0	Video stream format. 0=Brooktree 1=CCIR 656 2=Zoom Video 3=16bit VIP 4=TRANSPORT STREAM
CAP_HDWNS_DEC	26	0x1	Horizontal downscaler or decimator. 0=downscaler 1=decimator
CAP_VIDEO_IN_FORMAT	29	0x0	Input format. 0=YVYU422 1=VYUY422
VBI_HORZ_DOWN	31:30	0x0	0=Normal 1=x2 2=x4

Capture 0 configuration register.

CAP0_VIDEO_SYNC_TEST - RW - 32 bits - [MMReg:0x968]			
Field Name	**Bits**	**Default**	**Description**
CAP_TEST_VID_SOF	0	0x0	Start of field.
CAP_TEST_VID_EOF	1	0x0	End of field.
CAP_TEST_VID_EOL	2	0x0	End of line.

CAP_TEST_VID_FIELD	3	0x0	Odd/Even field. 0=Even Field 1=Odd Field
CAP_TEST_SYNC_EN	5	0x0	Test sync enable. 0=Normal 1=Test Mode

Capture port 0 sync test.

CAP0_ONESHOT_BUF_OFFSET - RW - 32 bits - [MMReg:0x96C]			
Field Name	**Bits**	**Default**	**Description**
CAP_ONESHOT_BUF_OFFSET	31:0	0x0	ONESHOT buffer starting address. NOTE: Bits 0:1 of this field are hardwired to ZERO.

ONESHOT buffer starting address.

CAP0_BUF_STATUS - R - 32 bits - [MMReg:0x970]			
Field Name	**Bits**	**Default**	**Description**
CAP_PRE_VID_BUF	1:0	0x0	Read only. Previous capture buffer.
CAP_CUR_VID_BUF	3:2	0x0	Read only. Current Capture buffer.
CAP_PRE_FIELD	4	0x0	Read only. Previous field.
CAP_CUR_FIELD	5	0x0	Read only. Current field.
CAP_PRE_VBI_BUF	7:6	0x0	Read only. Previous VBI buffer.
CAP_CUR_VBI_BUF	9:8	0x0	Read only. Current VBI buffer.
CAP_VBI_BUF_STATUS	10	0x0	Read only. VBI busy status. 0=done 1=busy
CAP_VIP_INC	28	0x0	Read only. Interlaced or not. 0=INTERLACED 1=NON_INTERLACED
CAP_VIP_PRE_REPEAT_FIELD	29	0x0	Read only. Previous buffer is new/repeat field. 0=new_field 1=repeated_field
CAP_CAP_BUF_STATUS	30	0x0	Read only. Capture buffer busy status. 0=done 1=busy

Capture 0 buffer status.

VID_BUFFER_CONTROL - RW - 32 bits - [MMReg:0x900]			
Field Name	**Bits**	**Default**	**Description**
CAP0_BUFFER_WATER_MARK	5:0	0x10	Capture 0 buffer water mark.
VID_BUFFER_RESET	20	0x0	Reset the buffer pointers. 0=NOT RESET 1=RESET
CAP_SWAP	22:21	0x0	Capture Port Swap control.
CAP0_BUFFER_EMPTY (R)	24	0x0	Capture 0's buffer empty status. 0=EMPTY 1=NOT EMPTY

Video Capture port buffer control.

VIDEOMUX_CNTL - RW - 32 bits - [MMReg:0x190]			
Field Name	**Bits**	**Default**	**Description**
ROM_CLK_DIVIDE	20:16	0x5	ROM clock divider
STR_ROMCLK	21	0x0	Extend ROM cycle
VIP_INTERNAL_DEBUG_SEL	24:22	0x0	
GEYSERVILLE_SENSE (R)	25	0x0	
GEYSERVILLE_INT_POL	26	0x0	

GPIO pin mux control

CAP_INT_STATUS - RW - 32 bits - [MMReg:0x90C]			
Field Name	**Bits**	**Default**	**Description**
CAP0_BUF0_INT (R)	0	0x0	Read only. Buffer 0 interrupt status. 0=No event 1=Event has occurred, interrupting if enabled
CAP0_BUF0_INT_AK (W)	0	0x0	Buf0 interrupt acknowledgment. 0=No effect 1=Clear status
CAP0_BUF0_EVEN_INT (R)	1	0x0	Read only. Buffer 0 even frame interrupt status. 0=No event 1=Event has occurred, interrupting if enabled

CAP0_BUF0_EVEN_INT_AK (W)	1	0x0	Buf0 even frame buffer interrupt acknowledgment. 0=No effect 1=Clear status
CAP0_BUF1_INT (R)	2	0x0	Read only. Buffer 1 interrupt status. 0=No event 1=Event has occurred, interrupting if enabled
CAP0_BUF1_INT_AK (W)	2	0x0	Buf1 interrupt acknowledgment. 0=No effect 1=Clear status
CAP0_BUF1_EVEN_INT (R)	3	0x0	Read only. Buffer 1 even frame interrupt status. 0=No event 1=Event has occurred, interrupting if enabled
CAP0_BUF1_EVEN_INT_AK (W)	3	0x0	Buf1 even frame buffer interrupt acknowledgment. 0=No effect 1=Clear status
CAP0_VBI0_INT (R)	4	0x0	Read only. VBI buffer 0 interrupt status. 0=No event 1=Event has occurred, interrupting if enabled
CAP0_VBI0_INT_AK (W)	4	0x0	VBI buffer 0 interrupt acknowledgment. 0=No effect 1=Clear status
CAP0_VBI1_INT (R)	5	0x0	Read only. VBI buffer 1 interrupt status. 0=No event 1=Event has occurred, interrupting if enabled
CAP0_VBI1_INT_AK (W)	5	0x0	VBI buffer 1 interrupt acknowledgment. 0=No effect 1=Clear status
CAP0_ONESHOT_INT (R)	6	0x0	Read only. ONESHOT buffer interrupt status. 0=No event 1=Event has occurred, interrupting if enabled
CAP0_ONESHOT_INT_AK (W)	6	0x0	ONESHOT buffer interrupt acknowledgment. 0=No effect 1=Clear status

Capture port interrupt control.

CAP_INT_CNTL - RW - 32 bits - [MMReg:0x908]			
Field Name	**Bits**	**Default**	**Description**
CAP0_BUF0_INT_EN	0	0x0	Capture 0 Buffer 0 Interrupt enable. 0=Disable 1=Enable

CAP0_BUF0_EVEN_INT_EN	1	0x0	Capture 0 Buffer 0 even frame Interrupt enable. 0=Disable 1=Enable
CAP0_BUF1_INT_EN	2	0x0	Capture 0 Buffer 1 Interrupt enable. 0=Disable 1=Enable
CAP0_BUF1_EVEN_INT_EN	3	0x0	Capture 0 Buffer 1 even frame Interrupt enable. 0=Disable 1=Enable
CAP0_VBI0_INT_EN	4	0x0	Capture 0 VBI Buffer 0 Interrupt enable. 0=Disable 1=Enable
CAP0_VBI1_INT_EN	5	0x0	Capture 0 VBI Buffer 1 Interrupt enable. 0=Disable 1=Enable
CAP0_ONESHOT_INT_EN	6	0x0	Capture 0 ONESHOT Buffer Interrupt enable. 0=Disable 1=Enable

Video Capture port interrupt control register

GEN_INT_CNTL - RW - 32 bits - [IOReg,MMReg:0x40]			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_MASK (DISPLAY)	0	0x0	Vertical blank interrupt mask. 0=Disable 1=Enable
CRTC_VLINE_MASK (DISPLAY)	1	0x0	Vertical line interrupt mask. 0=Disable 1=Enable
CRTC_VSYNC_MASK (DISPLAY)	2	0x0	Vertical sync interrupt mask. 0=Disable 1=Enable
SNAPSHOT_MASK (DISPLAY)	3	0x0	Snapshot interrupt mask. 0=Disable 1=Enable
FP_DETECT_MASK (DISPLAY)	4	0x0	Hot plug detect (HPD) interrupt mask. 0=Disable 1=Enable
CRTC2_VLINE_MASK (DISPLAY)	5	0x0	0=Disable 1=Enable

CRTC2_VSYNC_MASK (DISPLAY)	6	0x0	0=Disable 1=Enable
SNAPSHOT2_MASK (DISPLAY)	7	0x0	0=Disable 1=Enable
CRTC2_VBLANK_MASK (DISPLAY)	9	0x0	0=Disable 1=Enable
FP2_DETECT_MASK (DISPLAY)	10	0x0	0=Disable 1=Enable
VSYNC_DIFF_OVER_LIMIT_MASK (TVOUT)	11	0x0	0=Disable 1=Enable
GUI_IDLE_MASK (RBBM)	19	0x0	GUI idle interrupt mask. 0=Disable 1=Enable
SW_INT_EN (HDP)	25	0x0	Software interrupt mask. 0=Disable 1=Enable
GEYSERVILLE_MASK (VIP)	27	0x0	0=Disable 1=Enable
HDCP_AUTHORIZED_INT_MASK (DISPLAY)	28	0x0	0=Disable 1=Enable
DVI_I2C_INT_MASK (DISPLAY)	29	0x0	0=Disable 1=Enable
GUIDMA_MASK (CP)	30	0x0	GUI DMA channel interrupt mask. 0=Disable 1=Enable
VIDDMA_MASK (CP)	31	0x0	Video capture DMA channel interrupt mask. 0=Disable 1=Enable

General Interrupt Control register.
The MASK/EN fields control whether the respective status bits are enabled to drive the system interrupt pin. Even if enabled here, the interrupt line is not driven unless enabled in the PCI configuration space.

EXTERN_TRIG_CNTL - RW - 32 bits - [MMReg:0x1BC]			
Field Name	**Bits**	**Default**	**Description**
EXTERN_TRIG_CLR (W)	0	0x0	0=Write 0 has no affect. 1=Write 1 sets EXTERN_TRIG to 0. This can then be used with WAIT_UNTIL(19) to stall until external signal pulses.
EXTERN_TRIG_READ (R)	1	0x0	0=Read 1 indicates WAIT condition not active. 1=Read 0 indicates WAIT condition active.

No description available for this register.

2.46 GPIO Registers

CONFIG_XSTRAP - R - 32 bits - [IOReg,MMReg:0xE4]			
Field Name	**Bits**	**Default**	**Description**
VGA_DISABLE	0	0x0	VGA controller capability disable.
ENINTB	3	0x0	Enable Interrup.
AGPSKEW	7:6	0x0	AGP1x clock feedback phaseeadjustment wrt refclk.
X1CLK_SKEW	9:8	0x0	Clock phase adjustment between x1clk and x2clk.
ID_DISABLE	14	0x0	Shut down the chip by not responding to config cycles.
AP_SIZE	17:16	0x0	
ROMIDCFG	22:20	0x0	
BUSCFG	26:24	0x0	

Strap read back.

GPIOPAD_MASK - RW - 32 bits - [MMReg:0x198]			
Field Name	**Bits**	**Default**	**Description**
GPIO_MASK	13:0	0x0	
LTGIO_MASK	16:14	0x0	

No description available for this register.

GPIOPAD_A - RW - 32 bits - [MMReg:0x19C]			
Field Name	**Bits**	**Default**	**Description**
GPIO_A	13:0	0x0	
LTGIO_A	16:14	0x0	

No description available for this register.

GPIOPAD_EN - RW - 32 bits - [MMReg:0x1A0]			
Field Name	**Bits**	**Default**	**Description**
GPIO_EN	13:0	0x0	
LTGIO_EN	16:14	0x0	

No description available for this register.

GPIOPAD_Y - R - 32 bits - [MMReg:0x1A4]			
Field Name	**Bits**	**Default**	**Description**
GPIO_Y	13:0	0x0	
LTGIO_Y	16:14	0x0	

No description available for this register.

VIPPAD_STRENGTH - RW - 32 bits - [MMReg:0x194]			
Field Name	**Bits**	**Default**	**Description**
ZV_LCDDATA_STRENGTH	1:0	0x2	0=Minimum drive. ~74 ohms. 1=Matched impedence drive. ~63 ohms. 2=Optimal drive. ~47 ohms. 3=Maximum drive. ~42 ohms.
ZV_LCDCNTL_STRENGTH	9:8	0x2	0=Minimum drive. ~74 ohms. 1=Matched impedence drive. ~63 ohms. 2=Optimal drive. ~47 ohms. 3=Maximum drive. ~42 ohms.
GPIO_STRENGTH	13:12	0x2	0=Minimum drive. ~74 ohms. 1=Matched impedence drive. ~63 ohms. 2=Optimal drive. ~47 ohms. 3=Maximum drive. ~42 ohms.

No description available for this register.

ZV_LCDPAD_MASK - RW - 32 bits - [MMReg:0x1A8]			
Field Name	**Bits**	**Default**	**Description**
ZV_LCDDATA_MASK	23:0	0x0	
ZV_LCDCNTL_MASK	27:24	0x0	

No description available for this register.

ZV_LCDPAD_A - RW - 32 bits - [MMReg:0x1AC]			
Field Name	**Bits**	**Default**	**Description**
ZV_LCDDATA_A	23:0	0x0	
ZV_LCDCNTL_A	27:24	0x0	

No description available for this register.

ZV_LCDPAD_EN - RW - 32 bits - [MMReg:0x1B0]			
Field Name	**Bits**	**Default**	**Description**
ZV_LCDDATA_EN	23:0	0x0	
ZV_LCDCNTL_EN	27:24	0x0	

No description available for this register.

ZV_LCDPAD_Y - R - 32 bits - [MMReg:0x1B4]			
Field Name	**Bits**	**Default**	**Description**
ZV_LCDDATA_Y	23:0	0x0	
ZV_LCDCNTL_Y	27:24	0x0	

No description available for this register.

2.47 Scratch Registers

Various scratch and debug registers

BIOS_0_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x10]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	Scratch pad for BIOS information

Scratch pad for BIOS information

BIOS_1_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x14]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	Scratch pad for BIOS information

Scratch pad for BIOS information

BIOS_2_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x18]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	Scratch pad for BIOS information

Scratch pad for BIOS information

BIOS_3_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x1C]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	Scratch pad for BIOS information

Scratch pad for BIOS information

BIOS_4_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x20]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	

No description available for this register.

BIOS_5_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x24]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	

No description available for this register.

BIOS_6_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x28]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	

No description available for this register.

BIOS_7_SCRATCH - RW - 32 bits - [IOReg,MMReg:0x2C]			
Field Name	**Bits**	**Default**	**Description**
BIOS_SCRATCH	31:0	0x0	

No description available for this register.

MEDIA_0_SCRATCH - RW - 32 bits - [MMReg:0x1F0]			
Field Name	**Bits**	**Default**	**Description**
MEDIA_0_SCRATCH	31:0	0x0	

No description available for this register.

MEDIA_1_SCRATCH - RW - 32 bits - [MMReg:0x1F4]			
Field Name	**Bits**	**Default**	**Description**
MEDIA_1_SCRACH	31:0	0x0	

No description available for this register.

TEST_DEBUG_CNTL - RW - 32 bits - [MMReg:0x120]			
Field Name	**Bits**	**Default**	**Description**
TEST_DEBUG_OUT_EN	0	0x0	
TEST_DEBUG_IN_EN	1	0x0	
TEST_IDDQ_EN	2	0x0	
TEST_BLOCK_SEL	13:8	0x0	
TEST_ENABLE	15	0x0	
TEST_DELAY_IN	16	0x0	

Test bus control register.

TEST_DEBUG_MUX - RW - 32 bits - [MMReg:0x124]			
Field Name	**Bits**	**Default**	**Description**
TEST_DEBUG_SEL	5:0	0x0	0=bit0=RING_OSC_EN, bit1=DELAY_CHAIN_LENGTH
TEST_CLK0	12:8	0x0	
TEST_CLK0_INV	15	0x0	
TEST_CLK1	20:16	0x0	
TEST_CLK1_INV	23	0x0	

No description available for this register.

TEST_DEBUG_OUT - R - 32 bits - [MMReg:0x12C]			
Field Name	**Bits**	**Default**	**Description**
TEST_DEBUG_OUTR	11:0	0x0	

No description available for this register.

VIP_HW_DEBUG - RW - 32 bits - [MMReg:0x1CC]			
Field Name	**Bits**	**Default**	**Description**
VIP_HW_6_DEBUG	6	0x0	
VIP_HW_7_DEBUG	7	0x0	

VIP_HW_8_DEBUG	8	0x0	
VIP_HW_C_DEBUG	12	0x0	
VIP_HW_D_DEBUG	13	0x0	
VIP_HW_E_DEBUG	14	0x0	
VIP_HW_F_DEBUG	15	0x0	

No description available for this register.

2.48 ROM Registers

SEPROM_CNTL1 - RW - 32 bits - [MMReg:0x1C0]			
Field Name	**Bits**	**Default**	**Description**
WRITE_ENABLE	0	0x0	Set WRITE_ENABLE to be the command field
WRITE_DISABLE	1	0x0	Set WRITE_DISABLE to be the command field
READ_CONFIG	2	0x0	Set READ_CONFIG to be the command field
WRITE_CONFIG	3	0x0	Set WRITE_CONFIG to be the command field
READ_STATUS	4	0x0	Set READ_STATUS to be the command field
SECT_TO_SRAM	5	0x0	Set SECT_TO_SRAM to be the command field
READY_BUSY (R)	7	0x0	Status bit that reflects the status of the HOLD/ READY_BUSY bus
SEPROM_BUSY (R)	8	0x0	Status bit that indicates the status of the SPI state machine
BCNT_OVER_WTE_EN	9	0x0	This bit must be set to '1' for burst ROM write/read. This bit works coherently with the BYTE_CNT.
RB_MASKB	10	0x0	
SOFT_RESET	11	0x0	
STATE_IDLEb (R)	12	0x0	
BYTE_CNT	23:16	0xff	The BYTE_CNT works coherently with the BCNT_OVER_WTE_EN. Programming the BYTE_CNT has no effect if BCNT_OVER_WTE_EN is '0'. The BYTE_CNT can be programmed to tell the SPI state machine how many byte will be sent/read. BYTE_CNT = 0 means 1 byte will be sent ' ' BYTE_CNT = 255 means 256 byte will be sent
SCK_PRESCALE	31:24	0x4	This changes the SCK period with this funcition: SCK period = (SCK_PRESCALE + 1) * 2 * (SYSTEM CLOCK period)

First SPI Serial ROM Control register

SEPROM_CNTL2 - RW - 32 bits - [MMReg:0x1C4]			
Field Name	**Bits**	**Default**	**Description**
WAIT_CYCLE	7:0	0x5	
AUTO_ADDR_SAMPLE	8	0x1	
SEC_COMMAND	23:16	0x0	

Second SPI Serial ROM Control register

This page intentionally left blank.

Appendix A
Cross Referenced Index

A.1 All Registers Sorted by Name

Table A-1 All Registers Sorted by Name

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
BRUSH_DATA22	2-100
BRUSH_DATA23	2-100
BRUSH_DATA24	2-101
BRUSH_DATA25	2-101
BRUSH_DATA26	2-101
BRUSH_DATA27	2-101
BRUSH_DATA28	2-101
BRUSH_DATA29	2-102
BRUSH_DATA3	2-96
BRUSH_DATA30	2-102
BRUSH_DATA31	2-102
BRUSH_DATA32	2-102
BRUSH_DATA33	2-102
BRUSH_DATA34	2-103
BRUSH_DATA35	2-103
BRUSH_DATA36	2-103
BRUSH_DATA37	2-103
BRUSH_DATA38	2-103
BRUSH_DATA39	2-104
BRUSH_DATA4	2-97
BRUSH_DATA40	2-104
BRUSH_DATA41	2-104
BRUSH_DATA42	2-104
BRUSH_DATA43	2-104
BRUSH_DATA44	2-105
BRUSH_DATA45	2-105
BRUSH_DATA46	2-105
BRUSH_DATA47	2-105
BRUSH_DATA48	2-105
BRUSH_DATA49	2-106
BRUSH_DATA5	2-97
BRUSH_DATA50	2-106
BRUSH_DATA51	2-106
BRUSH_DATA52	2-106
BRUSH_DATA53	2-106
BRUSH_DATA54	2-107

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
BRUSH_DATA55	2-107
BRUSH_DATA56	2-107
BRUSH_DATA57	2-107
BRUSH_DATA58	2-107
BRUSH_DATA59	2-108
BRUSH_DATA6	2-97
BRUSH_DATA60	2-108
BRUSH_DATA61	2-108
BRUSH_DATA62	2-108
BRUSH_DATA63	2-108
BRUSH_DATA7	2-97
BRUSH_DATA8	2-97
BRUSH_DATA9	2-98
BRUSH_Y_X	2-95
BUS_CNTL	2-10
BUS_CNTL1	2-11
CACHE_LINE	2-8
CAP_INT_CNTL	2-383
CAP_INT_STATUS	2-382
CAP0_BUF_PITCH	2-376
CAP0_BUF_STATUS	2-381
CAP0_BUF0_EVEN_OFFSET	2-375
CAP0_BUF0_OFFSET	2-375
CAP0_BUF1_EVEN_OFFSET	2-376
CAP0_BUF1_OFFSET	2-375
CAP0_CONFIG	2-379
CAP0_DEBUG	2-378
CAP0_H_WINDOW	2-376
CAP0_ONESHOT_BUF_OFFSET	2-381
CAP0_PORT_MODE_CNTL	2-378
CAP0_TRIG_CNTL	2-378
CAP0_V_WINDOW	2-376
CAP0_VBI_H_WINDOW	2-377
CAP0_VBI_V_WINDOW	2-377
CAP0_VBI0_OFFSET	2-377
CAP0_VBI1_OFFSET	2-377
CAP0_VIDEO_SYNC_TEST	2-380

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
CAPABILITIES_PTR	2-6
CG_TEST_MACRO_RW_CNTL	2-56
CG_TEST_MACRO_RW_DATA	2-56
CG_TEST_MACRO_RW_READ	2-56
CG_TEST_MACRO_RW_WRITE	2-56
CLK_PIN_CNTL	2-33
CLK_PWRMGT_CNTL	2-30
CLK_PWRMGT_CNTL	2-49
CLOCK_CNTL_DATA	2-42
CLOCK_CNTL_DATA	2-57
CLOCK_CNTL_DATA	2-59
CLOCK_CNTL_DATA	2-274
CLOCK_CNTL_INDEX	2-42
CLOCK_CNTL_INDEX	2-57
CLOCK_CNTL_INDEX	2-59
CLOCK_CNTL_INDEX	2-274
CLR_CMP_CLR_DST	2-113
CLR_CMP_CLR_SRC	2-112
CLR_CMP_CNTL	2-112
CLR_CMP_MSK	2-113
COMMAND	2-2
CONFIG_APER_0_BASE	2-6
CONFIG_APER_1_BASE	2-6
CONFIG_APER_SIZE	2-7
CONFIG_CNTL	2-6
CONFIG_MEMSIZE	2-6
CONFIG_REG_1_BASE	2-7
CONFIG_REG_APER_SIZE	2-7
CONFIG_XSTRAP	2-387
CP_CSQ_ADDR	2-131
CP_CSQ_APER_INDIRECT	2-133
CP_CSQ_APER_PRIMARY	2-133
CP_CSQ_CNTL	2-129
CP_CSQ_DATA	2-131
CP_CSQ_STAT	2-132
CP_DEBUG	2-134

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
CRTC8_DATA	2-352
CRTC8_IDX	2-172
CRTC8_IDX	2-352
CUR_CLR0	2-255
CUR_CLR1	2-255
CUR_HORZ_VERT_OFF	2-254
CUR_HORZ_VERT_POSN	2-253
CUR_OFFSET	2-253
CUR2_CLR0	2-257
CUR2_CLR1	2-258
CUR2_HORZ_VERT_OFF	2-257
CUR2_HORZ_VERT_POSN	2-256
CUR2_OFFSET	2-255
DAC_CNTL	2-18
DAC_CNTL	2-151
DAC_CNTL2	2-157
DAC_CRC_SIG1	2-155
DAC_CRC_SIG2	2-156
DAC_CRC2_SIG1	2-156
DAC_CRC2_SIG2	2-157
DAC_DATA	2-154
DAC_EXT_CNTL	2-155
DAC_MACRO_CNTL	2-159
DAC_MASK	2-154
DAC_R_INDEX	2-154
DAC_W_INDEX	2-154
DEFAULT_PITCH_OFFSET	2-91
DEFAULT_SC_BOTTOM_RIGHT	2-119
DEFAULT2_PITCH_OFFSET	2-92
DEFAULT2_SC_BOTTOM_RIGHT	2-119
DEVICE_ID	2-2
DISP_HW_DEBUG	2-149
DISP_MERGE_CNTL	2-143
DISP_MISC_CNTL	2-144
DISP_OUTPUT_CNTL	2-142
DISP_PWR_MAN	2-147

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
DISP_TEST_DEBUG_CNTL	2-148
DISP_TEST_MACRO_RW_CNTL	2-301
DISP_TEST_MACRO_RW_DATA	2-301
DISP_TEST_MACRO_RW_READ	2-301
DISP_TEST_MACRO_RW_WRITE	2-301
DISP2_MERGE_CNTL	2-144
DISPLAY_BASE_ADDR	2-142
DMA_GUI_ACT_DSCRPTR	2-137
DMA_GUI_COMMAND	2-135
DMA_GUI_DST_ADDR	2-135
DMA_GUI_SRC_ADDR	2-135
DMA_GUI_STATUS	2-136
DMA_GUI_TABLE_ADDR	2-135
DP_BRUSH_BKGD_CLR	2-95
DP_BRUSH_FRGD_CLR	2-95
DP_CNTL	2-116
DP_CNTL_XDIR_YDIR_YMAJOR	2-96
DP_DATATYPE	2-117
DP_DST_ENDIAN	2-113
DP_GUI_MASTER_CNTL	2-93
DP_MIX	2-118
DP_SRC_BKGD_CLR	2-114
DP_SRC_ENDIAN	2-113
DP_SRC_FRGD_CLR	2-114
DP_WRITE_MSK	2-118
DP_XOP	2-122
DST_HEIGHT	2-90
DST_HEIGHT_WIDTH	2-93
DST_HEIGHT_WIDTH_8	2-109
DST_HEIGHT_Y	2-111
DST_LINE_END	2-114
DST_LINE_PATCOUNT	2-115
DST_LINE_START	2-114
DST_OFFSET	2-89
DST_PITCH	2-89
DST_PITCH_OFFSET	2-91
DST_TILE	2-119

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
DST_WIDTH	2-89
DST_WIDTH_HEIGHT	2-110
DST_WIDTH_X	2-109
DST_WIDTH_X_INCY	2-110
DST_X	2-90
DST_X_Y	2-110
DST_Y	2-90
DST_Y_X	2-92
DSTCACHE_CTLSTAT	2-123
DSTCACHE_MODE	2-123
DVI_I2C_CNTL_0	2-287
DVI_I2C_CNTL_1	2-288
DVI_I2C_DATA	2-288
EXT_MEM_CNTL	2-67
EXTERN_TRIG_CNTL	2-386
FCP_CNTL	2-372
FP_CRTC_H_TOTAL_DISP	2-299
FP_CRTC_V_TOTAL_DISP	2-300
FP_GEN_CNTL	2-290
FP_H_SYNC_STRT_WID	2-298
FP_H2_SYNC_STRT_WID	2-289
FP_HORZ_STRETCH	2-296
FP_HORZ_VERT_ACTIVE	2-299
FP_V_SYNC_STRT_WID	2-298
FP_V2_SYNC_STRT_WID	2-289
FP_VERT_STRETCH	2-297
FP2_GEN_CNTL	2-295
GEN_INT_CNTL	2-84
GEN_INT_CNTL	2-137
GEN_INT_CNTL	2-165
GEN_INT_CNTL	2-330
GEN_INT_CNTL	2-337
GEN_INT_CNTL	2-384
GEN_INT_STATUS	2-86
GEN_INT_STATUS	2-139
GEN_INT_STATUS	2-166

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
ISYNC_CNTL	2-80
LATENCY	2-8
LVDS_DIGTMDS_CRC	2-284
LVDS_GEN_CNTL	2-284
LVDS_PLL_CNTL	2-285
LVDS_SS_GEN_CNTL	2-286
M_SPLL_REF_FB_DIV	2-46
M_SPLL_REF_FB_DIV	2-57
MAX_LATENCY	2-8
MC_AGP_LOCATION	2-69
MC_DEBUG	2-74
MC_FB_LOCATION	2-69
MC_FB_LOCATION	2-345
MC_STATUS	2-76
MCLK_CNTL	2-54
MCLK_MISC	2-55
MDLL_CKO	2-53
MDLL_RDCKA	2-53
MEDIA_0_SCRATCH	2-391
MEDIA_1_SCRATCH	2-391
MEM_BASE	2-4
MEM_CNTL	2-66
MEM_INIT_LATENCY_TIMER	2-69
MEM_IO_CNTL_A0	2-71
MEM_IO_CNTL_A1	2-72
MEM_IO_OE_CNTL	2-74
MEM_SDRAM_MODE_REG	2-70
MEM_VGA_RP_SEL	2-336
MEM_VGA_WP_SEL	2-336
MIN_GRANT	2-9
MM_DATA	2-15
MM_INDEX	2-14
MPLL_CNTL	2-52
N_VIF_COUNT	2-307
N_VIF2_COUNT	2-308
NQWAIT_UNTIL	2-84
OV0_AUTO_FLIP_CNTRL	2-237

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
OV0_BASE_ADDR	2-234
OV0_DEINTERLACE_PATTERN	2-239
OV0_FILTER_CNTL	2-243
OV0_FLAG_CNTRL	2-246
OV0_FOUR_TAP_COEF_0	2-244
OV0_FOUR_TAP_COEF_1	2-244
OV0_FOUR_TAP_COEF_2	2-245
OV0_FOUR_TAP_COEF_3	2-245
OV0_FOUR_TAP_COEF_4	2-245
OV0_GAMMA_0_F	2-250
OV0_GAMMA_10_1F	2-250
OV0_GAMMA_20_3F	2-251
OV0_GAMMA_380_3BF	2-251
OV0_GAMMA_3C0_3FF	2-252
OV0_GAMMA_40_7F	2-251
OV0_GRPH_KEY_CLR_HIGH	2-247
OV0_GRPH_KEY_CLR_LOW	2-247
OV0_H_INC	2-240
OV0_KEY_CNTL	2-248
OV0_LIN_TRANS_A	2-249
OV0_LIN_TRANS_B	2-249
OV0_LIN_TRANS_C	2-249
OV0_LIN_TRANS_D	2-249
OV0_LIN_TRANS_E	2-250
OV0_LIN_TRANS_F	2-250
OV0_P1_BLANK_LINES_AT_TOP	2-233
OV0_P1_H_ACCUM_INIT	2-241
OV0_P1_V_ACCUM_INIT	2-232
OV0_P1_X_START_END	2-242
OV0_P2_X_START_END	2-242
OV0_P23_BLANK_LINES_AT_TOP	2-233
OV0_P23_H_ACCUM_INIT	2-242
OV0_P23_V_ACCUM_INIT	2-232
OV0_P3_X_START_END	2-243
OV0_PIPELINE_CNTL	2-227
OV0_REG_LOAD_CNTL	2-228

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
OV0_SCALE_CNTL	2-229
OV0_SLICE_CNTL	2-246
OV0_STEP_BY	2-240
OV0_SUBMIT_HISTORY	2-240
OV0_TEST	2-248
OV0_V_INC	2-232
OV0_VID_BUF_PITCH0_VALUE	2-236
OV0_VID_BUF_PITCH1_VALUE	2-237
OV0_VID_BUF0_BASE_ADRS	2-234
OV0_VID_BUF1_BASE_ADRS	2-235
OV0_VID_BUF2_BASE_ADRS	2-235
OV0_VID_BUF3_BASE_ADRS	2-235
OV0_VID_BUF4_BASE_ADRS	2-236
OV0_VID_BUF5_BASE_ADRS	2-236
OV0_VID_KEY_CLR_HIGH	2-247
OV0_VID_KEY_CLR_LOW	2-246
OV0_Y_X_END	2-227
OV0_Y_X_START	2-227
OV1_PIPELINE_CNTL	2-252
OV1_Y_X_END	2-252
OV1_Y_X_START	2-252
OVR_CLR	2-259
OVR_WID_LEFT_RIGHT	2-260
OVR_WID_TOP_BOTTOM	2-260
OVR2_CLR	2-259
OVR2_WID_LEFT_RIGHT	2-259
OVR2_WID_TOP_BOTTOM	2-259
P2PLL_CNTL	2-41
P2PLL_DIV_0	2-47
P2PLL_DIV_0	2-65
P2PLL_DIV_0	2-280
P2PLL_REF_DIV	2-41
PAD_AGPINPUT_DELAY	2-16
PAD_CTLR_STRENGTH	2-15
PAD_CTLR_UPDATE	2-16
PALETTE_30_DATA	2-303
PALETTE_DATA	2-303

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
RBBM_SOFT_RESET	2-78
RBBM_STATUS	2-79
REG_BASE	2-4
REGPROG_INF	2-8
REVISION_ID	2-3
RMX_HORZ_PHASE	2-304
SC_BOTTOM	2-116
SC_BOTTOM_RIGHT	2-118
SC_LEFT	2-115
SC_RIGHT	2-115
SC_TOP	2-115
SC_TOP_LEFT	2-118
SCLK_CNTL	2-36
SCRATCH_ADDR	2-130
SCRATCH_REG0	2-132
SCRATCH_REG1	2-132
SCRATCH_REG2	2-132
SCRATCH_REG3	2-133
SCRATCH_REG4	2-133
SCRATCH_REG5	2-133
SCRATCH_UMSK	2-129
SEPROM_CNTL1	2-394
SEPROM_CNTL2	2-395
SEQ00	2-170
SEQ01	2-170
SEQ02	2-350
SEQ03	2-171
SEQ04	2-351
SEQ8_DATA	2-171
SEQ8_DATA	2-350
SEQ8_IDX	2-171
SEQ8_IDX	2-350
SNAPSHOT_F_COUNT	2-307
SNAPSHOT_VH_COUNTS	2-307
SNAPSHOT_VIF_COUNT	2-307
SNAPSHOT2_F_COUNT	2-308
SNAPSHOT2_VH_COUNTS	2-308

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
SNAPSHOT2_VIF_COUNT	2-308
SPLL_CNTL	2-35
SRC_CLUT_ADDRESS	2-120
SRC_CLUT_DATA	2-120
SRC_CLUT_DATA_RD	2-120
SRC_OFFSET	2-111
SRC_PITCH	2-111
SRC_PITCH_OFFSET	2-91
SRC_SC_BOTTOM	2-116
SRC_SC_BOTTOM_RIGHT	2-119
SRC_SC_RIGHT	2-116
SRC_TILE	2-120
SRC_X	2-90
SRC_X_Y	2-109
SRC_Y	2-90
SRC_Y_X	2-92
STATUS	2-3
SUB_CLASS	2-5
SUBPIC_0_PAL	2-269
SUBPIC_1_PAL	2-269
SUBPIC_2_PAL	2-269
SUBPIC_3_PAL	2-269
SUBPIC_4_PAL	2-270
SUBPIC_5_PAL	2-270
SUBPIC_6_PAL	2-270
SUBPIC_7_PAL	2-270
SUBPIC_8_PAL	2-271
SUBPIC_9_PAL	2-271
SUBPIC_A_PAL	2-271
SUBPIC_B_PAL	2-271
SUBPIC_BTN_HLI_COLCON	2-268
SUBPIC_BTN_HLI_Y_X_END	2-268
SUBPIC_BTN_HLI_Y_X_START	2-268
SUBPIC_BUF0_OFFSET	2-266
SUBPIC_BUF1_OFFSET	2-267
SUBPIC_C_PAL	2-272

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
SUBPIC_CNTL	2-265
SUBPIC_D_PAL	2-272
SUBPIC_DEFCOLCON	2-265
SUBPIC_E_PAL	2-272
SUBPIC_F_PAL	2-272
SUBPIC_H_ACCUM_INIT	2-273
SUBPIC_H_INC	2-266
SUBPIC_LC0_OFFSET	2-267
SUBPIC_LC1_OFFSET	2-267
SUBPIC_PALETTE_DATA	2-269
SUBPIC_PALETTE_INDEX	2-268
SUBPIC_PITCH	2-267
SUBPIC_V_ACCUM_INIT	2-273
SUBPIC_V_INC	2-266
SUBPIC_Y_X_END	2-266
SUBPIC_Y_X_START	2-266
SURFACE_ACCESS_CLR	2-359
SURFACE_ACCESS_FLAGS	2-357
SURFACE_CNTL	2-357
SURFACE0_INFO	2-365
SURFACE0_LOWER_BOUND	2-361
SURFACE0_UPPER_BOUND	2-363
SURFACE1_INFO	2-366
SURFACE1_LOWER_BOUND	2-361
SURFACE1_UPPER_BOUND	2-363
SURFACE2_INFO	2-366
SURFACE2_LOWER_BOUND	2-361
SURFACE2_UPPER_BOUND	2-363
SURFACE3_INFO	2-367
SURFACE3_LOWER_BOUND	2-361
SURFACE3_UPPER_BOUND	2-363
SURFACE4_INFO	2-368
SURFACE4_LOWER_BOUND	2-362
SURFACE4_UPPER_BOUND	2-364
SURFACE5_INFO	2-369
SURFACE5_LOWER_BOUND	2-362
SURFACE5_UPPER_BOUND	2-364

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
SURFACE6_INFO	2-370
SURFACE6_LOWER_BOUND	2-362
SURFACE6_UPPER_BOUND	2-364
SURFACE7_INFO	2-370
SURFACE7_LOWER_BOUND	2-362
SURFACE7_UPPER_BOUND	2-364
SW_SEMAPHORE	2-336
TEST_DEBUG_CNTL	2-392
TEST_DEBUG_MUX	2-392
TEST_DEBUG_OUT	2-392
TMDS_CNTL	2-281
TMDS_CRC	2-282
TMDS_PATTERN_GEN_SEED	2-283
TMDS_PLL_CNTL	2-283
TMDS_SYNC_CHAR_SETA	2-281
TMDS_SYNC_CHAR_SETB	2-282
TMDS_TRANSMITTER_CNTL	2-282
TV_CRC_CNTL	2-324
TV_DAC_CNTL	2-323
TV_DTO_INCREMENTS	2-40
TV_FCOUNT	2-313
TV_FRESTART	2-314
TV_FTOTAL	2-313
TV_GAIN_LIMIT_SETTINGS	2-320
TV_HCOUNT	2-312
TV_HDISP	2-312
TV_HOST_RD_WT_CNTL	2-315
TV_HOST_READ_DATA	2-314
TV_HOST_WRITE_DATA	2-315
TV_HRESTART	2-314
TV_HSTART	2-312
TV_HTOTAL	2-312
TV_LINEAR_GAIN_SETTINGS	2-320
TV_MASTER_CNTL	2-309
TV_MODULATOR_CNTL1	2-320
TV_MODULATOR_CNTL2	2-321

Table A-1 All Registers Sorted by Name (Continued)

Table A-1 All Registers Sorted by Name (Continued)

Register Name	Page No.
ZV_LCDPAD_EN	*2-389*
ZV_LCDPAD_MASK	*2-388*
ZV_LCDPAD_Y	*2-389*

A.2 MMR Registers Sorted by Name

Table A-2 MMR Registers Sorted by Name

Register Name	Address	Secondary Address	Page
ADAPTER_ID	MMReg:0xF2C	pciConfig:0x2C	2-5
ADAPTER_ID_W	MMReg:0xF4C:R	pciConfig:0x4C	2-4
AGP_BASE	MMReg:0x170		2-66
AGP_CAP_ID	MMReg:0xF58	pciConfig:0x58	2-14
AGP_CNTL	MMReg:0x174		2-13
AGP_COMMAND	MMReg:0xF60:R	pciConfig:0x60	2-13
AGP_STATUS	MMReg:0xF5C	pciConfig:0x5C	2-14
AIC_CTRL	MMReg:0x1D0		2-25
AIC_HI_ADDR	MMReg:0x1E0		2-27
AIC_LO_ADDR	MMReg:0x1DC		2-26
AIC_PT_BASE	MMReg:0x1D8		2-26
AIC_STAT	MMReg:0x1D4		2-26
AIC_TLB_ADDR	MMReg:0x1E4		2-27
AIC_TLB_DATA	MMReg:0x1E8		2-27
AUX_WINDOW_HORZ_CNTL	MMReg:0x2D8		2-305
AUX_WINDOW_VERT_CNTL	MMReg:0x2DC		2-305
BASE_CODE	MMReg:0xF0B	pciConfig:0xB	2-4
BIOS_ROM	MMReg:0xF30:R	pciConfig:0x30	2-5
BIST	MMReg:0xF0F	pciConfig:0xF	2-5
BM_STATUS	MMReg:0x160		2-12
BRUSH_DATA0	MMReg:0x1480		2-96
BRUSH_DATA1	MMReg:0x1484		2-96
BRUSH_DATA10	MMReg:0x14A8		2-98
BRUSH_DATA11	MMReg:0x14AC		2-98
BRUSH_DATA12	MMReg:0x14B0		2-98
BRUSH_DATA13	MMReg:0x14B4		2-98
BRUSH_DATA14	MMReg:0x14B8		2-99
BRUSH_DATA15	MMReg:0x14BC		2-99
BRUSH_DATA16	MMReg:0x14C0		2-99
BRUSH_DATA17	MMReg:0x14C4		2-99
BRUSH_DATA18	MMReg:0x14C8		2-99
BRUSH_DATA19	MMReg:0x14CC		2-100
BRUSH_DATA2	MMReg:0x1488		2-96
BRUSH_DATA20	MMReg:0x14D0		2-100
BRUSH_DATA21	MMReg:0x14D4		2-100

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
BRUSH_DATA22	MMReg:0x14D8		2-100
BRUSH_DATA23	MMReg:0x14DC		2-100
BRUSH_DATA24	MMReg:0x14E0		2-101
BRUSH_DATA25	MMReg:0x14E4		2-101
BRUSH_DATA26	MMReg:0x14E8		2-101
BRUSH_DATA27	MMReg:0x14EC		2-101
BRUSH_DATA28	MMReg:0x14F0		2-101
BRUSH_DATA29	MMReg:0x14F4		2-102
BRUSH_DATA3	MMReg:0x148C		2-96
BRUSH_DATA30	MMReg:0x14F8		2-102
BRUSH_DATA31	MMReg:0x14FC		2-102
BRUSH_DATA32	MMReg:0x1500		2-102
BRUSH_DATA33	MMReg:0x1504		2-102
BRUSH_DATA34	MMReg:0x1508		2-103
BRUSH_DATA35	MMReg:0x150C		2-103
BRUSH_DATA36	MMReg:0x1510		2-103
BRUSH_DATA37	MMReg:0x1514		2-103
BRUSH_DATA38	MMReg:0x1518		2-103
BRUSH_DATA39	MMReg:0x151C		2-104
BRUSH_DATA4	MMReg:0x1490		2-97
BRUSH_DATA40	MMReg:0x1520		2-104
BRUSH_DATA41	MMReg:0x1524		2-104
BRUSH_DATA42	MMReg:0x1528		2-104
BRUSH_DATA43	MMReg:0x152C		2-104
BRUSH_DATA44	MMReg:0x1530		2-105
BRUSH_DATA45	MMReg:0x1534		2-105
BRUSH_DATA46	MMReg:0x1538		2-105
BRUSH_DATA47	MMReg:0x153C		2-105
BRUSH_DATA48	MMReg:0x1540		2-105
BRUSH_DATA49	MMReg:0x1544		2-106
BRUSH_DATA5	MMReg:0x1494		2-97
BRUSH_DATA50	MMReg:0x1548		2-106
BRUSH_DATA51	MMReg:0x154C		2-106
BRUSH_DATA52	MMReg:0x1550		2-106
BRUSH_DATA53	MMReg:0x1554		2-106
BRUSH_DATA54	MMReg:0x1558		2-107

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
BRUSH_DATA55	MMReg:0x155C		2-107
BRUSH_DATA56	MMReg:0x1560		2-107
BRUSH_DATA57	MMReg:0x1564		2-107
BRUSH_DATA58	MMReg:0x1568		2-107
BRUSH_DATA59	MMReg:0x156C		2-108
BRUSH_DATA6	MMReg:0x1498		2-97
BRUSH_DATA60	MMReg:0x1570		2-108
BRUSH_DATA61	MMReg:0x1574		2-108
BRUSH_DATA62	MMReg:0x1578		2-108
BRUSH_DATA63	MMReg:0x157C		2-108
BRUSH_DATA7	MMReg:0x149C		2-97
BRUSH_DATA8	MMReg:0x14A0		2-97
BRUSH_DATA9	MMReg:0x14A4		2-98
BRUSH_Y_X	MMReg:0x1474		2-95
CACHE_LINE	MMReg:0xF0C:R	pciConfig:0xC	2-8
CAP_INT_CNTL	MMReg:0x908		2-383
CAP_INT_STATUS	MMReg:0x90C		2-382
CAP0_BUF_PITCH	MMReg:0x930		2-376
CAP0_BUF_STATUS	MMReg:0x970		2-381
CAP0_BUF0_EVEN_OFFSET	MMReg:0x928		2-375
CAP0_BUF0_OFFSET	MMReg:0x920		2-375
CAP0_BUF1_EVEN_OFFSET	MMReg:0x92C		2-376
CAP0_BUF1_OFFSET	MMReg:0x924		2-375
CAP0_CONFIG	MMReg:0x958		2-379
CAP0_DEBUG	MMReg:0x954		2-378
CAP0_H_WINDOW	MMReg:0x938		2-376
CAP0_ONESHOT_BUF_OFFSET	MMReg:0x96C		2-381
CAP0_PORT_MODE_CNTL	MMReg:0x94C		2-378
CAP0_TRIG_CNTL	MMReg:0x950		2-378
CAP0_V_WINDOW	MMReg:0x934		2-376
CAP0_VBI_H_WINDOW	MMReg:0x948		2-377
CAP0_VBI_V_WINDOW	MMReg:0x944		2-377
CAP0_VBI0_OFFSET	MMReg:0x93C		2-377
CAP0_VBI1_OFFSET	MMReg:0x940		2-377
CAP0_VIDEO_SYNC_TEST	MMReg:0x968		2-380
CAPABILITIES_PTR	MMReg:0xF34	pciConfig:0x34	2-6
CLR_CMP_CLR_DST	MMReg:0x15C8		2-113

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
CLR_CMP_CLR_SRC	MMReg:0x15C4		2-112
CLR_CMP_CNTL	MMReg:0x15C0		2-112
CLR_CMP_MSK	MMReg:0x15CC		2-113
COMMAND	MMReg:0xF04:R	pciConfig:0x4	2-2
CONFIG_APER_0_BASE	MMReg:0x100		2-6
CONFIG_APER_1_BASE	MMReg:0x104		2-6
CONFIG_APER_SIZE	MMReg:0x108		2-7
CONFIG_REG_1_BASE	MMReg:0x10C		2-7
CONFIG_REG_APER_SIZE	MMReg:0x110		2-7
CP_CSQ_ADDR	MMReg:0x7F0		2-131
CP_CSQ_APER_INDIRECT	MMReg:0x13FC		2-133
CP_CSQ_APER_PRIMARY	MMReg:0x11FC		2-133
CP_CSQ_CNTL	MMReg:0x740		2-129
CP_CSQ_DATA	MMReg:0x7F4		2-131
CP_CSQ_STAT	MMReg:0x7F8		2-132
CP_DEBUG	MMReg:0x7EC		2-134
CP_IB_BASE	MMReg:0x738		2-128
CP_IB_BUFSZ	MMReg:0x73C		2-128
CP_ME_CNTL	MMReg:0x7D0		2-130
CP_ME_RAM_ADDR	MMReg:0x7D4		2-130
CP_ME_RAM_DATAH	MMReg:0x7DC		2-131
CP_ME_RAM_DATAL	MMReg:0x7E0		2-131
CP_ME_RAM_RADDR	MMReg:0x7D8		2-131
CP_RB_BASE	MMReg:0x700		2-127
CP_RB_CNTL	MMReg:0x704		2-126
CP_RB_RPTR	MMReg:0x710		2-127
CP_RB_RPTR_ADDR	MMReg:0x70C		2-127
CP_RB_RPTR_WR	MMReg:0x71C		2-127
CP_RB_WPTR	MMReg:0x714		2-128
CP_RB_WPTR_DELAY	MMReg:0x718		2-128
CP_STAT	MMReg:0x7C0		2-134
CRT_CRTC_H_SYNC_STRT_WID	MMReg:0x258		2-207
CRT_CRTC_V_SYNC_STRT_WID	MMReg:0x25C		2-208
CRTC_CRNT_FRAME	MMReg:0x214		2-202
CRTC_DEBUG	MMReg:0x21C		2-203
CRTC_GUI_TRIG_VLINE	MMReg:0x218		2-202

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
CRTC_H_SYNC_STRT_WID	MMReg:0x204		2-200
CRTC_H_TOTAL_DISP	MMReg:0x200		2-200
CRTC_MORE_CNTL	MMReg:0x27C		2-208
CRTC_OFFSET	MMReg:0x224		2-204
CRTC_OFFSET_CNTL	MMReg:0x228		2-205
CRTC_OFFSET_RIGHT	MMReg:0x220		2-203
CRTC_PITCH	MMReg:0x22C		2-207
CRTC_V_SYNC_STRT_WID	MMReg:0x20C		2-201
CRTC_V_TOTAL_DISP	MMReg:0x208		2-201
CRTC_VLINE_CRNT_VLINE	MMReg:0x210		2-202
CRTC2_CRNT_FRAME	MMReg:0x314		2-216
CRTC2_DEBUG	MMReg:0x31C		2-216
CRTC2_DISPLAY_BASE_ADDR	MMReg:0x33C		2-219
CRTC2_GEN_CNTL	MMReg:0x3F8		2-220
CRTC2_GUI_TRIG_VLINE	MMReg:0x318		2-216
CRTC2_H_SYNC_STRT_WID	MMReg:0x304		2-214
CRTC2_H_TOTAL_DISP	MMReg:0x300		2-214
CRTC2_OFFSET	MMReg:0x324		2-217
CRTC2_OFFSET_CNTL	MMReg:0x328		2-218
CRTC2_PITCH	MMReg:0x32C		2-219
CRTC2_STATUS	MMReg:0x3FC		2-222
CRTC2_V_SYNC_STRT_WID	MMReg:0x30C		2-215
CRTC2_V_TOTAL_DISP	MMReg:0x308		2-215
CRTC2_VLINE_CRNT_VLINE	MMReg:0x310		2-215
CUR_CLR0	MMReg:0x26C		2-255
CUR_CLR1	MMReg:0x270		2-255
CUR_HORZ_VERT_OFF	MMReg:0x268		2-254
CUR_HORZ_VERT_POSN	MMReg:0x264		2-253
CUR_OFFSET	MMReg:0x260		2-253
CUR2_CLR0	MMReg:0x36C		2-257
CUR2_CLR1	MMReg:0x370		2-258
CUR2_HORZ_VERT_OFF	MMReg:0x368		2-257
CUR2_HORZ_VERT_POSN	MMReg:0x364		2-256
CUR2_OFFSET	MMReg:0x360		2-255
DAC_CRC_SIG1	MMReg:0xD18		2-155
DAC_CRC_SIG2	MMReg:0xD1C		2-156
DAC_CRC2_SIG1	MMReg:0xD70		2-156

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
DAC_CRC2_SIG2	MMReg:0xD74		2-157
DAC_EXT_CNTL	MMReg:0x280		2-155
DAC_MACRO_CNTL	MMReg:0xD04		2-159
DEFAULT_PITCH_OFFSET	MMReg:0x16E0		2-91
DEFAULT_SC_BOTTOM_RIGHT	MMReg:0x16E8		2-119
DEFAULT2_PITCH_OFFSET	MMReg:0x16F8		2-92
DEFAULT2_SC_BOTTOM_RIGHT	MMReg:0x16DC		2-119
DEVICE_ID	MMReg:0xF02	pciConfig:0x2	2-2
DISP_HW_DEBUG	MMReg:0xD14		2-149
DISP_MERGE_CNTL	MMReg:0xD60		2-143
DISP_MISC_CNTL	MMReg:0xD00		2-144
DISP_OUTPUT_CNTL	MMReg:0xD64		2-142
DISP_PWR_MAN	MMReg:0xD08		2-147
DISP_TEST_DEBUG_CNTL	MMReg:0xD10		2-148
DISP2_MERGE_CNTL	MMReg:0xD68		2-144
DISPLAY_BASE_ADDR	MMReg:0x23C		2-142
DMA_GUI_ACT_DSCRPTR	MMReg:0x794		2-137
DMA_GUI_COMMAND	MMReg:0x78C		2-135
DMA_GUI_DST_ADDR	MMReg:0x788		2-135
DMA_GUI_SRC_ADDR	MMReg:0x784		2-135
DMA_GUI_STATUS	MMReg:0x790		2-136
DMA_GUI_TABLE_ADDR	MMReg:0x780		2-135
DP_BRUSH_BKGD_CLR	MMReg:0x1478		2-95
DP_BRUSH_FRGD_CLR	MMReg:0x147C		2-95
DP_CNTL	MMReg:0x16C0		2-116
DP_CNTL_XDIR_YDIR_YMAJOR	MMReg:0x16D0		2-96
DP_DATATYPE	MMReg:0x16C4		2-117
DP_DST_ENDIAN	MMReg:0x15D0		2-113
DP_GUI_MASTER_CNTL	MMReg:0x146C		2-93
DP_MIX	MMReg:0x16C8		2-118
DP_SRC_BKGD_CLR	MMReg:0x15DC		2-114
DP_SRC_ENDIAN	MMReg:0x15D4		2-113
DP_SRC_FRGD_CLR	MMReg:0x15D8		2-114
DP_WRITE_MSK	MMReg:0x16CC		2-118
DP_XOP	MMReg:0x17F8		2-122
DST_HEIGHT	MMReg:0x1410		2-90

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
DST_HEIGHT_WIDTH	MMReg:0x143C		2-93
DST_HEIGHT_WIDTH_8	MMReg:0x158C		2-109
DST_HEIGHT_Y	MMReg:0x15A0		2-111
DST_LINE_END	MMReg:0x1604		2-114
DST_LINE_PATCOUNT	MMReg:0x1608		2-115
DST_LINE_START	MMReg:0x1600		2-114
DST_OFFSET	MMReg:0x1404		2-89
DST_PITCH	MMReg:0x1408		2-89
DST_PITCH_OFFSET	MMReg:0x142C		2-91
DST_TILE	MMReg:0x1700		2-119
DST_WIDTH	MMReg:0x140C		2-89
DST_WIDTH_HEIGHT	MMReg:0x1598		2-110
DST_WIDTH_X	MMReg:0x1588		2-109
DST_WIDTH_X_INCY	MMReg:0x159C		2-110
DST_X	MMReg:0x141C		2-90
DST_X_Y	MMReg:0x1594		2-110
DST_Y	MMReg:0x1420		2-90
DST_Y_X	MMReg:0x1438		2-92
DSTCACHE_CTLSTAT	MMReg:0x1714		2-123
DSTCACHE_MODE	MMReg:0x1710		2-123
DVI_I2C_CNTL_0	MMReg:0x2E0		2-287
DVI_I2C_CNTL_1	MMReg:0x2E4		2-288
DVI_I2C_DATA	MMReg:0x2E8		2-288
EXT_MEM_CNTL	MMReg:0x144		2-67
EXTERN_TRIG_CNTL	MMReg:0x1BC		2-386
FCP_CNTL	MMReg:0x910		2-372
FP_CRTC_H_TOTAL_DISP	MMReg:0x250		2-299
FP_CRTC_V_TOTAL_DISP	MMReg:0x254		2-300
FP_GEN_CNTL	MMReg:0x284		2-290
FP_H_SYNC_STRT_WID	MMReg:0x2C4		2-298
FP_H2_SYNC_STRT_WID	MMReg:0x3C4		2-289
FP_HORZ_STRETCH	MMReg:0x28C		2-296
FP_HORZ_VERT_ACTIVE	MMReg:0x278		2-299
FP_V_SYNC_STRT_WID	MMReg:0x2C8		2-298
FP_V2_SYNC_STRT_WID	MMReg:0x3C8		2-289
FP_VERT_STRETCH	MMReg:0x290		2-297
FP2_GEN_CNTL	MMReg:0x288		2-295

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
GPIOPAD_A	MMReg:0x19C		2-387
GPIOPAD_EN	MMReg:0x1A0		2-388
GPIOPAD_MASK	MMReg:0x198		2-387
GPIOPAD_Y	MMReg:0x1A4		2-388
GRPH_BUFFER_CNTL	MMReg:0x2F0		2-163
GRPH2_BUFFER_CNTL	MMReg:0x3F0		2-164
HDP_DEBUG	MMReg:0x138		2-336
HEADER	MMReg:0xF0E	pciConfig:0xE	2-7
HOST_DATA_LAST	MMReg:0x17E0		2-122
HOST_DATA0	MMReg:0x17C0		2-120
HOST_DATA1	MMReg:0x17C4		2-121
HOST_DATA2	MMReg:0x17C8		2-121
HOST_DATA3	MMReg:0x17CC		2-121
HOST_DATA4	MMReg:0x17D0		2-121
HOST_DATA5	MMReg:0x17D4		2-122
HOST_DATA6	MMReg:0x17D8		2-122
HOST_DATA7	MMReg:0x17DC		2-122
HOST_PATH_CNTL	MMReg:0x130		2-335
ICON_CLR0	MMReg:0x2BC		2-262
ICON_CLR1	MMReg:0x2C0		2-262
ICON_HORZ_VERT_OFF	MMReg:0x2B8		2-262
ICON_HORZ_VERT_POSN	MMReg:0x2B4		2-261
ICON_OFFSET	MMReg:0x2B0		2-261
ICON2_CLR0	MMReg:0x3BC		2-264
ICON2_CLR1	MMReg:0x3C0		2-264
ICON2_HORZ_VERT_OFF	MMReg:0x3B8		2-263
ICON2_HORZ_VERT_POSN	MMReg:0x3B4		2-263
ICON2_OFFSET	MMReg:0x3B0		2-263
IDCT_AUTH	MMReg:0x1F8C		2-125
IDCT_AUTH_CONTROL	MMReg:0x1F88		2-125
IDCT_CONTROL	MMReg:0x1FBC		2-124
IDCT_LEVELS	MMReg:0x1F84		2-124
IDCT_RUNS	MMReg:0x1F80		2-124
INTERRUPT_LINE	MMReg:0xF3C:R	pciConfig:0x3C	2-7
INTERRUPT_PIN	MMReg:0xF3D	pciConfig:0x3D	2-8
IO_BASE	MMReg:0xF14:R	pciConfig:0x14	2-3

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
ISYNC_CNTL	MMReg:0x1724		2-80
LATENCY	MMReg:0xF0D:R	pciConfig:0xD	2-8
LVDS_DIGTMDS_CRC	MMReg:0x2CC		2-284
LVDS_GEN_CNTL	MMReg:0x2D0		2-284
LVDS_PLL_CNTL	MMReg:0x2D4		2-285
LVDS_SS_GEN_CNTL	MMReg:0x2EC		2-286
MAX_LATENCY	MMReg:0xF3F	pciConfig:0x3F	2-8
MC_AGP_LOCATION	MMReg:0x14C		2-69
MC_DEBUG	MMReg:0x188		2-74
MC_FB_LOCATION	MMReg:0x148		2-69
MC_FB_LOCATION	MMReg:0x148		2-345
MC_STATUS	MMReg:0x150		2-76
MEDIA_0_SCRATCH	MMReg:0x1F0		2-391
MEDIA_1_SCRATCH	MMReg:0x1F4		2-391
MEM_BASE	MMReg:0xF10:R	pciConfig:0x10	2-4
MEM_CNTL	MMReg:0x140		2-66
MEM_INIT_LATENCY_TIMER	MMReg:0x154		2-69
MEM_IO_CNTL_A0	MMReg:0x178		2-71
MEM_IO_CNTL_A1	MMReg:0x17C		2-72
MEM_IO_OE_CNTL	MMReg:0x18C		2-74
MEM_SDRAM_MODE_REG	MMReg:0x158		2-70
MIN_GRANT	MMReg:0xF3E	pciConfig:0x3E	2-9
N_VIF_COUNT	MMReg:0x248		2-307
N_VIF2_COUNT	MMReg:0x348		2-308
NQWAIT_UNTIL	MMReg:0xE50		2-84
OV0_AUTO_FLIP_CNTRL	MMReg:0x470		2-237
OV0_BASE_ADDR	MMReg:0x43C		2-234
OV0_DEINTERLACE_PATTERN	MMReg:0x474		2-239
OV0_FILTER_CNTL	MMReg:0x4A0		2-243
OV0_FLAG_CNTRL	MMReg:0x4DC		2-246
OV0_FOUR_TAP_COEF_0	MMReg:0x4B0		2-244
OV0_FOUR_TAP_COEF_1	MMReg:0x4B4		2-244
OV0_FOUR_TAP_COEF_2	MMReg:0x4B8		2-245
OV0_FOUR_TAP_COEF_3	MMReg:0x4BC		2-245
OV0_FOUR_TAP_COEF_4	MMReg:0x4C0		2-245
OV0_GAMMA_0_F	MMReg:0xD40		2-250
OV0_GAMMA_10_1F	MMReg:0xD44		2-250

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
OV0_GAMMA_20_3F	MMReg:0xD48		2-251
OV0_GAMMA_380_3BF	MMReg:0xD50		2-251
OV0_GAMMA_3C0_3FF	MMReg:0xD54		2-252
OV0_GAMMA_40_7F	MMReg:0xD4C		2-251
OV0_GRPH_KEY_CLR_HIGH	MMReg:0x4F0		2-247
OV0_GRPH_KEY_CLR_LOW	MMReg:0x4EC		2-247
OV0_H_INC	MMReg:0x480		2-240
OV0_KEY_CNTL	MMReg:0x4F4		2-248
OV0_LIN_TRANS_A	MMReg:0xD20		2-249
OV0_LIN_TRANS_B	MMReg:0xD24		2-249
OV0_LIN_TRANS_C	MMReg:0xD28		2-249
OV0_LIN_TRANS_D	MMReg:0xD2C		2-249
OV0_LIN_TRANS_E	MMReg:0xD30		2-250
OV0_LIN_TRANS_F	MMReg:0xD34		2-250
OV0_P1_BLANK_LINES_AT_TOP	MMReg:0x430		2-233
OV0_P1_H_ACCUM_INIT	MMReg:0x488		2-241
OV0_P1_V_ACCUM_INIT	MMReg:0x428		2-232
OV0_P1_X_START_END	MMReg:0x494		2-242
OV0_P2_X_START_END	MMReg:0x498		2-242
OV0_P23_BLANK_LINES_AT_TOP	MMReg:0x434		2-233
OV0_P23_H_ACCUM_INIT	MMReg:0x48C		2-242
OV0_P23_V_ACCUM_INIT	MMReg:0x42C		2-232
OV0_P3_X_START_END	MMReg:0x49C		2-243
OV0_PIPELINE_CNTL	MMReg:0x408		2-227
OV0_REG_LOAD_CNTL	MMReg:0x410		2-228
OV0_SCALE_CNTL	MMReg:0x420		2-229
OV0_SLICE_CNTL	MMReg:0x4E0		2-246
OV0_STEP_BY	MMReg:0x484		2-240
OV0_SUBMIT_HISTORY	MMReg:0x478		2-240
OV0_TEST	MMReg:0x4F8		2-248
OV0_V_INC	MMReg:0x424		2-232
OV0_VID_BUF_PITCH0_VALUE	MMReg:0x460		2-236
OV0_VID_BUF_PITCH1_VALUE	MMReg:0x464		2-237
OV0_VID_BUF0_BASE_ADRS	MMReg:0x440		2-234
OV0_VID_BUF1_BASE_ADRS	MMReg:0x444		2-235
OV0_VID_BUF2_BASE_ADRS	MMReg:0x448		2-235

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
OV0_VID_BUF3_BASE_ADRS	MMReg:0x44C		2-235
OV0_VID_BUF4_BASE_ADRS	MMReg:0x450		2-236
OV0_VID_BUF5_BASE_ADRS	MMReg:0x454		2-236
OV0_VID_KEY_CLR_HIGH	MMReg:0x4E8		2-247
OV0_VID_KEY_CLR_LOW	MMReg:0x4E4		2-246
OV0_Y_X_END	MMReg:0x404		2-227
OV0_Y_X_START	MMReg:0x400		2-227
OV1_PIPELINE_CNTL	MMReg:0x608		2-252
OV1_Y_X_END	MMReg:0x604		2-252
OV1_Y_X_START	MMReg:0x600		2-252
OVR_CLR	MMReg:0x230		2-259
OVR_WID_LEFT_RIGHT	MMReg:0x234		2-260
OVR_WID_TOP_BOTTOM	MMReg:0x238		2-260
OVR2_CLR	MMReg:0x330		2-259
OVR2_WID_LEFT_RIGHT	MMReg:0x334		2-259
OVR2_WID_TOP_BOTTOM	MMReg:0x338		2-259
PAD_AGPINPUT_DELAY	MMReg:0x164		2-16
PAD_CTLR_STRENGTH	MMReg:0x168		2-15
PAD_CTLR_UPDATE	MMReg:0x16C		2-16
PD2_DATA	MMReg:0x1718		2-123
PM_STATUS	MMReg:0xF54:R	pciConfig:0x54	2-28
PMI_CAP_ID	MMReg:0xF50	pciConfig:0x50	2-28
PMI_DATA	MMReg:0xF57	pciConfig:0x57	2-29
PMI_NXT_CAP_PTR	MMReg:0xF51	pciConfig:0x51	2-28
PMI_PMC_REG	MMReg:0xF52	pciConfig:0x52	2-28
RBBM_CMDFIFO_ADDR	MMReg:0xE70		2-81
RBBM_CMDFIFO_DATAH	MMReg:0xE78		2-82
RBBM_CMDFIFO_DATAL	MMReg:0xE74		2-82
RBBM_CMDFIFO_STAT	MMReg:0xE7C		2-82
RBBM_DEBUG	MMReg:0xE6C		2-84
RBBM_GUICNTL	MMReg:0x172C		2-81
RBBM_STATUS	MMReg:0xE40	MMReg:0x1740	2-79
REG_BASE	MMReg:0xF18:R	pciConfig:0x18	2-4
REGPROG_INF	MMReg:0xF09	pciConfig:0x9	2-8
REVISION_ID	MMReg:0xF08	pciConfig:0x8	2-3
RMX_HORZ_PHASE	MMReg:0xDBC		2-304
SC_BOTTOM	MMReg:0x164C		2-116

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
SC_BOTTOM_RIGHT	MMReg:0x16F0		2-118
SC_LEFT	MMReg:0x1640		2-115
SC_RIGHT	MMReg:0x1644		2-115
SC_TOP	MMReg:0x1648		2-115
SC_TOP_LEFT	MMReg:0x16EC		2-118
SCRATCH_ADDR	MMReg:0x774		2-130
SCRATCH_REG0	MMReg:0x15E0		2-132
SCRATCH_REG1	MMReg:0x15E4		2-132
SCRATCH_REG2	MMReg:0x15E8		2-132
SCRATCH_REG3	MMReg:0x15EC		2-133
SCRATCH_REG4	MMReg:0x15F0		2-133
SCRATCH_REG5	MMReg:0x15F4		2-133
SCRATCH_UMSK	MMReg:0x770		2-129
SEPROM_CNTL1	MMReg:0x1C0		2-394
SEPROM_CNTL2	MMReg:0x1C4		2-395
SNAPSHOT_F_COUNT	MMReg:0x244		2-307
SNAPSHOT_VH_COUNTS	MMReg:0x240		2-307
SNAPSHOT_VIF_COUNT	MMReg:0x24C		2-307
SNAPSHOT2_F_COUNT	MMReg:0x344		2-308
SNAPSHOT2_VH_COUNTS	MMReg:0x340		2-308
SNAPSHOT2_VIF_COUNT	MMReg:0x34C		2-308
SRC_CLUT_ADDRESS	MMReg:0x1780		2-120
SRC_CLUT_DATA	MMReg:0x1784		2-120
SRC_CLUT_DATA_RD	MMReg:0x1788		2-120
SRC_OFFSET	MMReg:0x15AC		2-111
SRC_PITCH	MMReg:0x15B0		2-111
SRC_PITCH_OFFSET	MMReg:0x1428		2-91
SRC_SC_BOTTOM	MMReg:0x165C		2-116
SRC_SC_BOTTOM_RIGHT	MMReg:0x16F4		2-119
SRC_SC_RIGHT	MMReg:0x1654		2-116
SRC_TILE	MMReg:0x1704		2-120
SRC_X	MMReg:0x1414		2-90
SRC_X_Y	MMReg:0x1590		2-109
SRC_Y	MMReg:0x1418		2-90
SRC_Y_X	MMReg:0x1434		2-92
STATUS	MMReg:0xF06:R	pciConfig:0x6	2-3

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
SUB_CLASS	MMReg:0xF0A	pciConfig:0xA	2-5
SUBPIC_BTN_HLI_COLCON	MMReg:0x570		2-268
SUBPIC_BTN_HLI_Y_X_END	MMReg:0x578		2-268
SUBPIC_BTN_HLI_Y_X_START	MMReg:0x574		2-268
SUBPIC_BUF0_OFFSET	MMReg:0x55C		2-266
SUBPIC_BUF1_OFFSET	MMReg:0x560		2-267
SUBPIC_CNTL	MMReg:0x540		2-265
SUBPIC_DEFCOLCON	MMReg:0x544		2-265
SUBPIC_H_ACCUM_INIT	MMReg:0x584		2-273
SUBPIC_H_INC	MMReg:0x558		2-266
SUBPIC_LC0_OFFSET	MMReg:0x564		2-267
SUBPIC_LC1_OFFSET	MMReg:0x568		2-267
SUBPIC_PALETTE_DATA	MMReg:0x580		2-269
SUBPIC_PALETTE_INDEX	MMReg:0x57C		2-268
SUBPIC_PITCH	MMReg:0x56C		2-267
SUBPIC_V_ACCUM_INIT	MMReg:0x588		2-273
SUBPIC_V_INC	MMReg:0x554		2-266
SUBPIC_Y_X_END	MMReg:0x550		2-266
SUBPIC_Y_X_START	MMReg:0x54C		2-266
SURFACE_ACCESS_CLR	MMReg:0xBFC		2-359
SURFACE_ACCESS_FLAGS	MMReg:0xBF8		2-357
SURFACE_CNTL	MMReg:0xB00		2-357
SURFACE0_INFO	MMReg:0xB0C		2-365
SURFACE0_LOWER_BOUND	MMReg:0xB04		2-361
SURFACE0_UPPER_BOUND	MMReg:0xB08		2-363
SURFACE1_INFO	MMReg:0xB1C		2-366
SURFACE1_LOWER_BOUND	MMReg:0xB14		2-361
SURFACE1_UPPER_BOUND	MMReg:0xB18		2-363
SURFACE2_INFO	MMReg:0xB2C		2-366
SURFACE2_LOWER_BOUND	MMReg:0xB24		2-361
SURFACE2_UPPER_BOUND	MMReg:0xB28		2-363
SURFACE3_INFO	MMReg:0xB3C		2-367
SURFACE3_LOWER_BOUND	MMReg:0xB34		2-361
SURFACE3_UPPER_BOUND	MMReg:0xB38		2-363
SURFACE4_INFO	MMReg:0xB4C		2-368
SURFACE4_LOWER_BOUND	MMReg:0xB44		2-362
SURFACE4_UPPER_BOUND	MMReg:0xB48		2-364

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
SURFACE5_INFO	MMReg:0xB5C		2-369
SURFACE5_LOWER_BOUND	MMReg:0xB54		2-362
SURFACE5_UPPER_BOUND	MMReg:0xB58		2-364
SURFACE6_INFO	MMReg:0xB6C		2-370
SURFACE6_LOWER_BOUND	MMReg:0xB64		2-362
SURFACE6_UPPER_BOUND	MMReg:0xB68		2-364
SURFACE7_INFO	MMReg:0xB7C		2-370
SURFACE7_LOWER_BOUND	MMReg:0xB74		2-362
SURFACE7_UPPER_BOUND	MMReg:0xB78		2-364
SW_SEMAPHORE	MMReg:0x13C		2-336
TEST_DEBUG_CNTL	MMReg:0x120		2-392
TEST_DEBUG_MUX	MMReg:0x124		2-392
TEST_DEBUG_OUT	MMReg:0x12C		2-392
TMDS_CNTL	MMReg:0x294		2-281
TMDS_CRC	MMReg:0x2A0		2-282
TMDS_PATTERN_GEN_SEED	MMReg:0x2AC		2-283
TMDS_PLL_CNTL	MMReg:0x2A8		2-283
TMDS_SYNC_CHAR_SETA	MMReg:0x298		2-281
TMDS_SYNC_CHAR_SETB	MMReg:0x29C		2-282
TMDS_TRANSMITTER_CNTL	MMReg:0x2A4		2-282
TV_CRC_CNTL	MMReg:0x890		2-324
TV_DAC_CNTL	MMReg:0x88C		2-323
TV_FCOUNT	MMReg:0x830		2-313
TV_FRESTART	MMReg:0x834		2-314
TV_FTOTAL	MMReg:0x82C		2-313
TV_GAIN_LIMIT_SETTINGS	MMReg:0x868		2-320
TV_HCOUNT	MMReg:0x81C		2-312
TV_HDISP	MMReg:0x810		2-312
TV_HOST_RD_WT_CNTL	MMReg:0x848		2-315
TV_HOST_READ_DATA	MMReg:0x840		2-314
TV_HOST_WRITE_DATA	MMReg:0x844		2-315
TV_HRESTART	MMReg:0x838		2-314
TV_HSTART	MMReg:0x818		2-312
TV_HTOTAL	MMReg:0x80C		2-312
TV_LINEAR_GAIN_SETTINGS	MMReg:0x86C		2-320
TV_MASTER_CNTL	MMReg:0x800		2-309

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Address	Page
TV_MODULATOR_CNTL1	MMReg:0x870		2-320
TV_MODULATOR_CNTL2	MMReg:0x874		2-321
TV_PRE_DAC_MUX_CNTL	MMReg:0x888		2-322
TV_RGB_CNTL	MMReg:0x804		2-310
TV_SYNC_CNTL	MMReg:0x808		2-311
TV_TIMING_CNTL	MMReg:0x850		2-316
TV_UPSAMP_AND_GAIN_CNTL	MMReg:0x864		2-319
TV_UV_ADR	MMReg:0x8AC		2-327
TV_VBI_20BIT_CNTL	MMReg:0x8A0		2-326
TV_VBI_CC_CNTL	MMReg:0x898		2-324
TV_VBI_DTO_CNTL	MMReg:0x8A4		2-326
TV_VBI_EDS_CNTL	MMReg:0x89C		2-325
TV_VBI_LEVEL_CNTL	MMReg:0x8A8		2-327
TV_VCOUNT	MMReg:0x828		2-313
TV_VDISP	MMReg:0x824		2-313
TV_VIDEO_PORT_SIG	MMReg:0x894		2-324
TV_VRESTART	MMReg:0x83C		2-314
TV_VSCALER_CNTL1	MMReg:0x84C		2-315
TV_VSCALER_CNTL2	MMReg:0x854		2-317
TV_VSYNC_DIFF_CNTL	MMReg:0x8F4		2-328
TV_VSYNC_DIFF_LIMITS	MMReg:0x8F8		2-329
TV_VSYNC_DIFF_RD_DATA	MMReg:0x8FC		2-329
TV_VTOTAL	MMReg:0x820		2-313
TV_Y_FALL_CNTL	MMReg:0x858		2-317
TV_Y_RISE_CNTL	MMReg:0x85C		2-318
TV_Y_SAW_TOOTH_CNTL	MMReg:0x860		2-318
VENDOR_ID	MMReg:0xF00	pciConfig:0x0	2-2
VGA_BUFFER_CNTL	MMReg:0x2F4		2-164
VID_BUFFER_CONTROL	MMReg:0x900		2-382
VIDEOMUX_CNTL	MMReg:0x190		2-382
VIP_HW_DEBUG	MMReg:0x1CC		2-392
VIPPAD_STRENGTH	MMReg:0x194		2-388
WAIT_UNTIL	MMReg:0x1720		2-82
ZV_LCDPAD_A	MMReg:0x1AC		2-389
ZV_LCDPAD_EN	MMReg:0x1B0		2-389
ZV_LCDPAD_MASK	MMReg:0x1A8		2-388
ZV_LCDPAD_Y	MMReg:0x1B4		2-389

A.3 MMR Registers Sorted by Address

Table A-3 MMR Registers Sorted by Address

Register Name	Address	Secondary Address	Page
CONFIG_APER_0_BASE	MMReg:0x100		2-6
CONFIG_APER_1_BASE	MMReg:0x104		2-6
CONFIG_APER_SIZE	MMReg:0x108		2-7
CONFIG_REG_1_BASE	MMReg:0x10C		2-7
CONFIG_REG_APER_SIZE	MMReg:0x110		2-7
CP_CSQ_APER_PRIMARY	MMReg:0x11FC		2-133
TEST_DEBUG_CNTL	MMReg:0x120		2-392
TEST_DEBUG_MUX	MMReg:0x124		2-392
TEST_DEBUG_OUT	MMReg:0x12C		2-392
HOST_PATH_CNTL	MMReg:0x130		2-335
HDP_DEBUG	MMReg:0x138		2-336
SW_SEMAPHORE	MMReg:0x13C		2-336
CP_CSQ_APER_INDIRECT	MMReg:0x13FC		2-133
MEM_CNTL	MMReg:0x140		2-66
DST_OFFSET	MMReg:0x1404		2-89
DST_PITCH	MMReg:0x1408		2-89
DST_WIDTH	MMReg:0x140C		2-89
DST_HEIGHT	MMReg:0x1410		2-90
SRC_X	MMReg:0x1414		2-90
SRC_Y	MMReg:0x1418		2-90
DST_X	MMReg:0x141C		2-90
DST_Y	MMReg:0x1420		2-90
SRC_PITCH_OFFSET	MMReg:0x1428		2-91
DST_PITCH_OFFSET	MMReg:0x142C		2-91
SRC_Y_X	MMReg:0x1434		2-92
DST_Y_X	MMReg:0x1438		2-92
DST_HEIGHT_WIDTH	MMReg:0x143C		2-93
EXT_MEM_CNTL	MMReg:0x144		2-67
DP_GUI_MASTER_CNTL	MMReg:0x146C		2-93
BRUSH_Y_X	MMReg:0x1474		2-95
DP_BRUSH_BKGD_CLR	MMReg:0x1478		2-95
DP_BRUSH_FRGD_CLR	MMReg:0x147C		2-95
MC_FB_LOCATION	MMReg:0x148		2-69
MC_FB_LOCATION	MMReg:0x148		2-345

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
BRUSH_DATA0	MMReg:0x1480		2-96
BRUSH_DATA1	MMReg:0x1484		2-96
BRUSH_DATA2	MMReg:0x1488		2-96
BRUSH_DATA3	MMReg:0x148C		2-96
BRUSH_DATA4	MMReg:0x1490		2-97
BRUSH_DATA5	MMReg:0x1494		2-97
BRUSH_DATA6	MMReg:0x1498		2-97
BRUSH_DATA7	MMReg:0x149C		2-97
BRUSH_DATA8	MMReg:0x14A0		2-97
BRUSH_DATA9	MMReg:0x14A4		2-98
BRUSH_DATA10	MMReg:0x14A8		2-98
BRUSH_DATA11	MMReg:0x14AC		2-98
BRUSH_DATA12	MMReg:0x14B0		2-98
BRUSH_DATA13	MMReg:0x14B4		2-98
BRUSH_DATA14	MMReg:0x14B8		2-99
BRUSH_DATA15	MMReg:0x14BC		2-99
MC_AGP_LOCATION	MMReg:0x14C		2-69
BRUSH_DATA16	MMReg:0x14C0		2-99
BRUSH_DATA17	MMReg:0x14C4		2-99
BRUSH_DATA18	MMReg:0x14C8		2-99
BRUSH_DATA19	MMReg:0x14CC		2-100
BRUSH_DATA20	MMReg:0x14D0		2-100
BRUSH_DATA21	MMReg:0x14D4		2-100
BRUSH_DATA22	MMReg:0x14D8		2-100
BRUSH_DATA23	MMReg:0x14DC		2-100
BRUSH_DATA24	MMReg:0x14E0		2-101
BRUSH_DATA25	MMReg:0x14E4		2-101
BRUSH_DATA26	MMReg:0x14E8		2-101
BRUSH_DATA27	MMReg:0x14EC		2-101
BRUSH_DATA28	MMReg:0x14F0		2-101
BRUSH_DATA29	MMReg:0x14F4		2-102
BRUSH_DATA30	MMReg:0x14F8		2-102
BRUSH_DATA31	MMReg:0x14FC		2-102
MC_STATUS	MMReg:0x150		2-76
BRUSH_DATA32	MMReg:0x1500		2-102
BRUSH_DATA33	MMReg:0x1504		2-102
BRUSH_DATA34	MMReg:0x1508		2-103

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
BRUSH_DATA35	MMReg:0x150C		2-103
BRUSH_DATA36	MMReg:0x1510		2-103
BRUSH_DATA37	MMReg:0x1514		2-103
BRUSH_DATA38	MMReg:0x1518		2-103
BRUSH_DATA39	MMReg:0x151C		2-104
BRUSH_DATA40	MMReg:0x1520		2-104
BRUSH_DATA41	MMReg:0x1524		2-104
BRUSH_DATA42	MMReg:0x1528		2-104
BRUSH_DATA43	MMReg:0x152C		2-104
BRUSH_DATA44	MMReg:0x1530		2-105
BRUSH_DATA45	MMReg:0x1534		2-105
BRUSH_DATA46	MMReg:0x1538		2-105
BRUSH_DATA47	MMReg:0x153C		2-105
MEM_INIT_LATENCY_TIMER	MMReg:0x154		2-69
BRUSH_DATA48	MMReg:0x1540		2-105
BRUSH_DATA49	MMReg:0x1544		2-106
BRUSH_DATA50	MMReg:0x1548		2-106
BRUSH_DATA51	MMReg:0x154C		2-106
BRUSH_DATA52	MMReg:0x1550		2-106
BRUSH_DATA53	MMReg:0x1554		2-106
BRUSH_DATA54	MMReg:0x1558		2-107
BRUSH_DATA55	MMReg:0x155C		2-107
BRUSH_DATA56	MMReg:0x1560		2-107
BRUSH_DATA57	MMReg:0x1564		2-107
BRUSH_DATA58	MMReg:0x1568		2-107
BRUSH_DATA59	MMReg:0x156C		2-108
BRUSH_DATA60	MMReg:0x1570		2-108
BRUSH_DATA61	MMReg:0x1574		2-108
BRUSH_DATA62	MMReg:0x1578		2-108
BRUSH_DATA63	MMReg:0x157C		2-108
MEM_SDRAM_MODE_REG	MMReg:0x158		2-70
DST_WIDTH_X	MMReg:0x1588		2-109
DST_HEIGHT_WIDTH_8	MMReg:0x158C		2-109
SRC_X_Y	MMReg:0x1590		2-109
DST_X_Y	MMReg:0x1594		2-110
DST_WIDTH_HEIGHT	MMReg:0x1598		2-110

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
DST_WIDTH_X_INCY	MMReg:0x159C		2-110
DST_HEIGHT_Y	MMReg:0x15A0		2-111
SRC_OFFSET	MMReg:0x15AC		2-111
SRC_PITCH	MMReg:0x15B0		2-111
CLR_CMP_CNTL	MMReg:0x15C0		2-112
CLR_CMP_CLR_SRC	MMReg:0x15C4		2-112
CLR_CMP_CLR_DST	MMReg:0x15C8		2-113
CLR_CMP_MSK	MMReg:0x15CC		2-113
DP_DST_ENDIAN	MMReg:0x15D0		2-113
DP_SRC_ENDIAN	MMReg:0x15D4		2-113
DP_SRC_FRGD_CLR	MMReg:0x15D8		2-114
DP_SRC_BKGD_CLR	MMReg:0x15DC		2-114
SCRATCH_REG0	MMReg:0x15E0		2-132
SCRATCH_REG1	MMReg:0x15E4		2-132
SCRATCH_REG2	MMReg:0x15E8		2-132
SCRATCH_REG3	MMReg:0x15EC		2-133
SCRATCH_REG4	MMReg:0x15F0		2-133
SCRATCH_REG5	MMReg:0x15F4		2-133
BM_STATUS	MMReg:0x160		2-12
DST_LINE_START	MMReg:0x1600		2-114
DST_LINE_END	MMReg:0x1604		2-114
DST_LINE_PATCOUNT	MMReg:0x1608		2-115
PAD_AGPINPUT_DELAY	MMReg:0x164		2-16
SC_LEFT	MMReg:0x1640		2-115
SC_RIGHT	MMReg:0x1644		2-115
SC_TOP	MMReg:0x1648		2-115
SC_BOTTOM	MMReg:0x164C		2-116
SRC_SC_RIGHT	MMReg:0x1654		2-116
SRC_SC_BOTTOM	MMReg:0x165C		2-116
PAD_CTLR_STRENGTH	MMReg:0x168		2-15
PAD_CTLR_UPDATE	MMReg:0x16C		2-16
DP_CNTL	MMReg:0x16C0		2-116
DP_DATATYPE	MMReg:0x16C4		2-117
DP_MIX	MMReg:0x16C8		2-118
DP_WRITE_MSK	MMReg:0x16CC		2-118
DP_CNTL_XDIR_YDIR_YMAJOR	MMReg:0x16D0		2-96
DEFAULT2_SC_BOTTOM_RIGHT	MMReg:0x16DC		2-119

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
DEFAULT_PITCH_OFFSET	MMReg:0x16E0		2-91
DEFAULT_SC_BOTTOM_RIGHT	MMReg:0x16E8		2-119
SC_TOP_LEFT	MMReg:0x16EC		2-118
SC_BOTTOM_RIGHT	MMReg:0x16F0		2-118
SRC_SC_BOTTOM_RIGHT	MMReg:0x16F4		2-119
DEFAULT2_PITCH_OFFSET	MMReg:0x16F8		2-92
AGP_BASE	MMReg:0x170		2-66
DST_TILE	MMReg:0x1700		2-119
SRC_TILE	MMReg:0x1704		2-120
DSTCACHE_MODE	MMReg:0x1710		2-123
DSTCACHE_CTLSTAT	MMReg:0x1714		2-123
PD2_DATA	MMReg:0x1718		2-123
WAIT_UNTIL	MMReg:0x1720		2-82
ISYNC_CNTL	MMReg:0x1724		2-80
RBBM_GUICNTL	MMReg:0x172C		2-81
AGP_CNTL	MMReg:0x174		2-13
MEM_IO_CNTL_A0	MMReg:0x178		2-71
SRC_CLUT_ADDRESS	MMReg:0x1780		2-120
SRC_CLUT_DATA	MMReg:0x1784		2-120
SRC_CLUT_DATA_RD	MMReg:0x1788		2-120
MEM_IO_CNTL_A1	MMReg:0x17C		2-72
HOST_DATA0	MMReg:0x17C0		2-120
HOST_DATA1	MMReg:0x17C4		2-121
HOST_DATA2	MMReg:0x17C8		2-121
HOST_DATA3	MMReg:0x17CC		2-121
HOST_DATA4	MMReg:0x17D0		2-121
HOST_DATA5	MMReg:0x17D4		2-122
HOST_DATA6	MMReg:0x17D8		2-122
HOST_DATA7	MMReg:0x17DC		2-122
HOST_DATA_LAST	MMReg:0x17E0		2-122
DP_XOP	MMReg:0x17F8		2-122
MC_DEBUG	MMReg:0x188		2-74
MEM_IO_OE_CNTL	MMReg:0x18C		2-74
VIDEOMUX_CNTL	MMReg:0x190		2-382
VIPPAD_STRENGTH	MMReg:0x194		2-388
GPIOPAD_MASK	MMReg:0x198		2-387

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
GPIOPAD_A	MMReg:0x19C		2-387
GPIOPAD_EN	MMReg:0x1A0		2-388
GPIOPAD_Y	MMReg:0x1A4		2-388
ZV_LCDPAD_MASK	MMReg:0x1A8		2-388
ZV_LCDPAD_A	MMReg:0x1AC		2-389
ZV_LCDPAD_EN	MMReg:0x1B0		2-389
ZV_LCDPAD_Y	MMReg:0x1B4		2-389
EXTERN_TRIG_CNTL	MMReg:0x1BC		2-386
SEPROM_CNTL1	MMReg:0x1C0		2-394
SEPROM_CNTL2	MMReg:0x1C4		2-395
VIP_HW_DEBUG	MMReg:0x1CC		2-392
AIC_CTRL	MMReg:0x1D0		2-25
AIC_STAT	MMReg:0x1D4		2-26
AIC_PT_BASE	MMReg:0x1D8		2-26
AIC_LO_ADDR	MMReg:0x1DC		2-26
AIC_HI_ADDR	MMReg:0x1E0		2-27
AIC_TLB_ADDR	MMReg:0x1E4		2-27
AIC_TLB_DATA	MMReg:0x1E8		2-27
MEDIA_0_SCRATCH	MMReg:0x1F0		2-391
MEDIA_1_SCRATCH	MMReg:0x1F4		2-391
IDCT_RUNS	MMReg:0x1F80		2-124
IDCT_LEVELS	MMReg:0x1F84		2-124
IDCT_AUTH_CONTROL	MMReg:0x1F88		2-125
IDCT_AUTH	MMReg:0x1F8C		2-125
IDCT_CONTROL	MMReg:0x1FBC		2-124
CRTC_H_TOTAL_DISP	MMReg:0x200		2-200
CRTC_H_SYNC_STRT_WID	MMReg:0x204		2-200
CRTC_V_TOTAL_DISP	MMReg:0x208		2-201
CRTC_V_SYNC_STRT_WID	MMReg:0x20C		2-201
CRTC_VLINE_CRNT_VLINE	MMReg:0x210		2-202
CRTC_CRNT_FRAME	MMReg:0x214		2-202
CRTC_GUI_TRIG_VLINE	MMReg:0x218		2-202
CRTC_DEBUG	MMReg:0x21C		2-203
CRTC_OFFSET_RIGHT	MMReg:0x220		2-203
CRTC_OFFSET	MMReg:0x224		2-204
CRTC_OFFSET_CNTL	MMReg:0x228		2-205
CRTC_PITCH	MMReg:0x22C		2-207

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
OVR_CLR	MMReg:0x230		2-259
OVR_WID_LEFT_RIGHT	MMReg:0x234		2-260
OVR_WID_TOP_BOTTOM	MMReg:0x238		2-260
DISPLAY_BASE_ADDR	MMReg:0x23C		2-142
SNAPSHOT_VH_COUNTS	MMReg:0x240		2-307
SNAPSHOT_F_COUNT	MMReg:0x244		2-307
N_VIF_COUNT	MMReg:0x248		2-307
SNAPSHOT_VIF_COUNT	MMReg:0x24C		2-307
FP_CRTC_H_TOTAL_DISP	MMReg:0x250		2-299
FP_CRTC_V_TOTAL_DISP	MMReg:0x254		2-300
CRT_CRTC_H_SYNC_STRT_WID	MMReg:0x258		2-207
CRT_CRTC_V_SYNC_STRT_WID	MMReg:0x25C		2-208
CUR_OFFSET	MMReg:0x260		2-253
CUR_HORZ_VERT_POSN	MMReg:0x264		2-253
CUR_HORZ_VERT_OFF	MMReg:0x268		2-254
CUR_CLR0	MMReg:0x26C		2-255
CUR_CLR1	MMReg:0x270		2-255
FP_HORZ_VERT_ACTIVE	MMReg:0x278		2-299
CRTC_MORE_CNTL	MMReg:0x27C		2-208
DAC_EXT_CNTL	MMReg:0x280		2-155
FP_GEN_CNTL	MMReg:0x284		2-290
FP2_GEN_CNTL	MMReg:0x288		2-295
FP_HORZ_STRETCH	MMReg:0x28C		2-296
FP_VERT_STRETCH	MMReg:0x290		2-297
TMDS_CNTL	MMReg:0x294		2-281
TMDS_SYNC_CHAR_SETA	MMReg:0x298		2-281
TMDS_SYNC_CHAR_SETB	MMReg:0x29C		2-282
TMDS_CRC	MMReg:0x2A0		2-282
TMDS_TRANSMITTER_CNTL	MMReg:0x2A4		2-282
TMDS_PLL_CNTL	MMReg:0x2A8		2-283
TMDS_PATTERN_GEN_SEED	MMReg:0x2AC		2-283
ICON_OFFSET	MMReg:0x2B0		2-261
ICON_HORZ_VERT_POSN	MMReg:0x2B4		2-261
ICON_HORZ_VERT_OFF	MMReg:0x2B8		2-262
ICON_CLR0	MMReg:0x2BC		2-262
ICON_CLR1	MMReg:0x2C0		2-262

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
FP_H_SYNC_STRT_WID	MMReg:0x2C4		2-298
FP_V_SYNC_STRT_WID	MMReg:0x2C8		2-298
LVDS_DIGTMDS_CRC	MMReg:0x2CC		2-284
LVDS_GEN_CNTL	MMReg:0x2D0		2-284
LVDS_PLL_CNTL	MMReg:0x2D4		2-285
AUX_WINDOW_HORZ_CNTL	MMReg:0x2D8		2-305
AUX_WINDOW_VERT_CNTL	MMReg:0x2DC		2-305
DVI_I2C_CNTL_0	MMReg:0x2E0		2-287
DVI_I2C_CNTL_1	MMReg:0x2E4		2-288
DVI_I2C_DATA	MMReg:0x2E8		2-288
LVDS_SS_GEN_CNTL	MMReg:0x2EC		2-286
GRPH_BUFFER_CNTL	MMReg:0x2F0		2-163
VGA_BUFFER_CNTL	MMReg:0x2F4		2-164
CRTC2_H_TOTAL_DISP	MMReg:0x300		2-214
CRTC2_H_SYNC_STRT_WID	MMReg:0x304		2-214
CRTC2_V_TOTAL_DISP	MMReg:0x308		2-215
CRTC2_V_SYNC_STRT_WID	MMReg:0x30C		2-215
CRTC2_VLINE_CRNT_VLINE	MMReg:0x310		2-215
CRTC2_CRNT_FRAME	MMReg:0x314		2-216
CRTC2_GUI_TRIG_VLINE	MMReg:0x318		2-216
CRTC2_DEBUG	MMReg:0x31C		2-216
CRTC2_OFFSET	MMReg:0x324		2-217
CRTC2_OFFSET_CNTL	MMReg:0x328		2-218
CRTC2_PITCH	MMReg:0x32C		2-219
OVR2_CLR	MMReg:0x330		2-259
OVR2_WID_LEFT_RIGHT	MMReg:0x334		2-259
OVR2_WID_TOP_BOTTOM	MMReg:0x338		2-259
CRTC2_DISPLAY_BASE_ADDR	MMReg:0x33C		2-219
SNAPSHOT2_VH_COUNTS	MMReg:0x340		2-308
SNAPSHOT2_F_COUNT	MMReg:0x344		2-308
N_VIF2_COUNT	MMReg:0x348		2-308
SNAPSHOT2_VIF_COUNT	MMReg:0x34C		2-308
CUR2_OFFSET	MMReg:0x360		2-255
CUR2_HORZ_VERT_POSN	MMReg:0x364		2-256
CUR2_HORZ_VERT_OFF	MMReg:0x368		2-257
CUR2_CLR0	MMReg:0x36C		2-257
CUR2_CLR1	MMReg:0x370		2-258

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
ICON2_OFFSET	MMReg:0x3B0		2-263
ICON2_HORZ_VERT_POSN	MMReg:0x3B4		2-263
ICON2_HORZ_VERT_OFF	MMReg:0x3B8		2-263
ICON2_CLR0	MMReg:0x3BC		2-264
ICON2_CLR1	MMReg:0x3C0		2-264
FP_H2_SYNC_STRT_WID	MMReg:0x3C4		2-289
FP_V2_SYNC_STRT_WID	MMReg:0x3C8		2-289
GRPH2_BUFFER_CNTL	MMReg:0x3F0		2-164
CRTC2_GEN_CNTL	MMReg:0x3F8		2-220
CRTC2_STATUS	MMReg:0x3FC		2-222
OV0_Y_X_START	MMReg:0x400		2-227
OV0_Y_X_END	MMReg:0x404		2-227
OV0_PIPELINE_CNTL	MMReg:0x408		2-227
OV0_REG_LOAD_CNTL	MMReg:0x410		2-228
OV0_SCALE_CNTL	MMReg:0x420		2-229
OV0_V_INC	MMReg:0x424		2-232
OV0_P1_V_ACCUM_INIT	MMReg:0x428		2-232
OV0_P23_V_ACCUM_INIT	MMReg:0x42C		2-232
OV0_P1_BLANK_LINES_AT_TOP	MMReg:0x430		2-233
OV0_P23_BLANK_LINES_AT_TOP	MMReg:0x434		2-233
OV0_BASE_ADDR	MMReg:0x43C		2-234
OV0_VID_BUF0_BASE_ADRS	MMReg:0x440		2-234
OV0_VID_BUF1_BASE_ADRS	MMReg:0x444		2-235
OV0_VID_BUF2_BASE_ADRS	MMReg:0x448		2-235
OV0_VID_BUF3_BASE_ADRS	MMReg:0x44C		2-235
OV0_VID_BUF4_BASE_ADRS	MMReg:0x450		2-236
OV0_VID_BUF5_BASE_ADRS	MMReg:0x454		2-236
OV0_VID_BUF_PITCH0_VALUE	MMReg:0x460		2-236
OV0_VID_BUF_PITCH1_VALUE	MMReg:0x464		2-237
OV0_AUTO_FLIP_CNTRL	MMReg:0x470		2-237
OV0_DEINTERLACE_PATTERN	MMReg:0x474		2-239
OV0_SUBMIT_HISTORY	MMReg:0x478		2-240
OV0_H_INC	MMReg:0x480		2-240
OV0_STEP_BY	MMReg:0x484		2-240
OV0_P1_H_ACCUM_INIT	MMReg:0x488		2-241
OV0_P23_H_ACCUM_INIT	MMReg:0x48C		2-242

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
OV0_P1_X_START_END	MMReg:0x494		2-242
OV0_P2_X_START_END	MMReg:0x498		2-242
OV0_P3_X_START_END	MMReg:0x49C		2-243
OV0_FILTER_CNTL	MMReg:0x4A0		2-243
OV0_FOUR_TAP_COEF_0	MMReg:0x4B0		2-244
OV0_FOUR_TAP_COEF_1	MMReg:0x4B4		2-244
OV0_FOUR_TAP_COEF_2	MMReg:0x4B8		2-245
OV0_FOUR_TAP_COEF_3	MMReg:0x4BC		2-245
OV0_FOUR_TAP_COEF_4	MMReg:0x4C0		2-245
OV0_FLAG_CNTRL	MMReg:0x4DC		2-246
OV0_SLICE_CNTL	MMReg:0x4E0		2-246
OV0_VID_KEY_CLR_LOW	MMReg:0x4E4		2-246
OV0_VID_KEY_CLR_HIGH	MMReg:0x4E8		2-247
OV0_GRPH_KEY_CLR_LOW	MMReg:0x4EC		2-247
OV0_GRPH_KEY_CLR_HIGH	MMReg:0x4F0		2-247
OV0_KEY_CNTL	MMReg:0x4F4		2-248
OV0_TEST	MMReg:0x4F8		2-248
SUBPIC_CNTL	MMReg:0x540		2-265
SUBPIC_DEFCOLCON	MMReg:0x544		2-265
SUBPIC_Y_X_START	MMReg:0x54C		2-266
SUBPIC_Y_X_END	MMReg:0x550		2-266
SUBPIC_V_INC	MMReg:0x554		2-266
SUBPIC_H_INC	MMReg:0x558		2-266
SUBPIC_BUF0_OFFSET	MMReg:0x55C		2-266
SUBPIC_BUF1_OFFSET	MMReg:0x560		2-267
SUBPIC_LC0_OFFSET	MMReg:0x564		2-267
SUBPIC_LC1_OFFSET	MMReg:0x568		2-267
SUBPIC_PITCH	MMReg:0x56C		2-267
SUBPIC_BTN_HLI_COLCON	MMReg:0x570		2-268
SUBPIC_BTN_HLI_Y_X_START	MMReg:0x574		2-268
SUBPIC_BTN_HLI_Y_X_END	MMReg:0x578		2-268
SUBPIC_PALETTE_INDEX	MMReg:0x57C		2-268
SUBPIC_PALETTE_DATA	MMReg:0x580		2-269
SUBPIC_H_ACCUM_INIT	MMReg:0x584		2-273
SUBPIC_V_ACCUM_INIT	MMReg:0x588		2-273
OV1_Y_X_START	MMReg:0x600		2-252
OV1_Y_X_END	MMReg:0x604		2-252

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
OV1_PIPELINE_CNTL	MMReg:0x608		2-252
CP_RB_BASE	MMReg:0x700		2-127
CP_RB_CNTL	MMReg:0x704		2-126
CP_RB_RPTR_ADDR	MMReg:0x70C		2-127
CP_RB_RPTR	MMReg:0x710		2-127
CP_RB_WPTR	MMReg:0x714		2-128
CP_RB_WPTR_DELAY	MMReg:0x718		2-128
CP_RB_RPTR_WR	MMReg:0x71C		2-127
CP_IB_BASE	MMReg:0x738		2-128
CP_IB_BUFSZ	MMReg:0x73C		2-128
CP_CSQ_CNTL	MMReg:0x740		2-129
SCRATCH_UMSK	MMReg:0x770		2-129
SCRATCH_ADDR	MMReg:0x774		2-130
DMA_GUI_TABLE_ADDR	MMReg:0x780		2-135
DMA_GUI_SRC_ADDR	MMReg:0x784		2-135
DMA_GUI_DST_ADDR	MMReg:0x788		2-135
DMA_GUI_COMMAND	MMReg:0x78C		2-135
DMA_GUI_STATUS	MMReg:0x790		2-136
DMA_GUI_ACT_DSCRPTR	MMReg:0x794		2-137
CP_STAT	MMReg:0x7C0		2-134
CP_ME_CNTL	MMReg:0x7D0		2-130
CP_ME_RAM_ADDR	MMReg:0x7D4		2-130
CP_ME_RAM_RADDR	MMReg:0x7D8		2-131
CP_ME_RAM_DATAH	MMReg:0x7DC		2-131
CP_ME_RAM_DATAL	MMReg:0x7E0		2-131
CP_DEBUG	MMReg:0x7EC		2-134
CP_CSQ_ADDR	MMReg:0x7F0		2-131
CP_CSQ_DATA	MMReg:0x7F4		2-131
CP_CSQ_STAT	MMReg:0x7F8		2-132
TV_MASTER_CNTL	MMReg:0x800		2-309
TV_RGB_CNTL	MMReg:0x804		2-310
TV_SYNC_CNTL	MMReg:0x808		2-311
TV_HTOTAL	MMReg:0x80C		2-312
TV_HDISP	MMReg:0x810		2-312
TV_HSTART	MMReg:0x818		2-312
TV_HCOUNT	MMReg:0x81C		2-312

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
TV_VTOTAL	MMReg:0x820		2-313
TV_VDISP	MMReg:0x824		2-313
TV_VCOUNT	MMReg:0x828		2-313
TV_FTOTAL	MMReg:0x82C		2-313
TV_FCOUNT	MMReg:0x830		2-313
TV_FRESTART	MMReg:0x834		2-314
TV_HRESTART	MMReg:0x838		2-314
TV_VRESTART	MMReg:0x83C		2-314
TV_HOST_READ_DATA	MMReg:0x840		2-314
TV_HOST_WRITE_DATA	MMReg:0x844		2-315
TV_HOST_RD_WT_CNTL	MMReg:0x848		2-315
TV_VSCALER_CNTL1	MMReg:0x84C		2-315
TV_TIMING_CNTL	MMReg:0x850		2-316
TV_VSCALER_CNTL2	MMReg:0x854		2-317
TV_Y_FALL_CNTL	MMReg:0x858		2-317
TV_Y_RISE_CNTL	MMReg:0x85C		2-318
TV_Y_SAW_TOOTH_CNTL	MMReg:0x860		2-318
TV_UPSAMP_AND_GAIN_CNTL	MMReg:0x864		2-319
TV_GAIN_LIMIT_SETTINGS	MMReg:0x868		2-320
TV_LINEAR_GAIN_SETTINGS	MMReg:0x86C		2-320
TV_MODULATOR_CNTL1	MMReg:0x870		2-320
TV_MODULATOR_CNTL2	MMReg:0x874		2-321
TV_PRE_DAC_MUX_CNTL	MMReg:0x888		2-322
TV_DAC_CNTL	MMReg:0x88C		2-323
TV_CRC_CNTL	MMReg:0x890		2-324
TV_VIDEO_PORT_SIG	MMReg:0x894		2-324
TV_VBI_CC_CNTL	MMReg:0x898		2-324
TV_VBI_EDS_CNTL	MMReg:0x89C		2-325
TV_VBI_20BIT_CNTL	MMReg:0x8A0		2-326
TV_VBI_DTO_CNTL	MMReg:0x8A4		2-326
TV_VBI_LEVEL_CNTL	MMReg:0x8A8		2-327
TV_UV_ADR	MMReg:0x8AC		2-327
TV_VSYNC_DIFF_CNTL	MMReg:0x8F4		2-328
TV_VSYNC_DIFF_LIMITS	MMReg:0x8F8		2-329
TV_VSYNC_DIFF_RD_DATA	MMReg:0x8FC		2-329
VID_BUFFER_CONTROL	MMReg:0x900		2-382
CAP_INT_CNTL	MMReg:0x908		2-383

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
CAP_INT_STATUS	MMReg:0x90C		2-382
FCP_CNTL	MMReg:0x910		2-372
CAP0_BUF0_OFFSET	MMReg:0x920		2-375
CAP0_BUF1_OFFSET	MMReg:0x924		2-375
CAP0_BUF0_EVEN_OFFSET	MMReg:0x928		2-375
CAP0_BUF1_EVEN_OFFSET	MMReg:0x92C		2-376
CAP0_BUF_PITCH	MMReg:0x930		2-376
CAP0_V_WINDOW	MMReg:0x934		2-376
CAP0_H_WINDOW	MMReg:0x938		2-376
CAP0_VBI0_OFFSET	MMReg:0x93C		2-377
CAP0_VBI1_OFFSET	MMReg:0x940		2-377
CAP0_VBI_V_WINDOW	MMReg:0x944		2-377
CAP0_VBI_H_WINDOW	MMReg:0x948		2-377
CAP0_PORT_MODE_CNTL	MMReg:0x94C		2-378
CAP0_TRIG_CNTL	MMReg:0x950		2-378
CAP0_DEBUG	MMReg:0x954		2-378
CAP0_CONFIG	MMReg:0x958		2-379
CAP0_VIDEO_SYNC_TEST	MMReg:0x968		2-380
CAP0_ONESHOT_BUF_OFFSET	MMReg:0x96C		2-381
CAP0_BUF_STATUS	MMReg:0x970		2-381
SURFACE_CNTL	MMReg:0xB00		2-357
SURFACE0_LOWER_BOUND	MMReg:0xB04		2-361
SURFACE0_UPPER_BOUND	MMReg:0xB08		2-363
SURFACE0_INFO	MMReg:0xB0C		2-365
SURFACE1_LOWER_BOUND	MMReg:0xB14		2-361
SURFACE1_UPPER_BOUND	MMReg:0xB18		2-363
SURFACE1_INFO	MMReg:0xB1C		2-366
SURFACE2_LOWER_BOUND	MMReg:0xB24		2-361
SURFACE2_UPPER_BOUND	MMReg:0xB28		2-363
SURFACE2_INFO	MMReg:0xB2C		2-366
SURFACE3_LOWER_BOUND	MMReg:0xB34		2-361
SURFACE3_UPPER_BOUND	MMReg:0xB38		2-363
SURFACE3_INFO	MMReg:0xB3C		2-367
SURFACE4_LOWER_BOUND	MMReg:0xB44		2-362
SURFACE4_UPPER_BOUND	MMReg:0xB48		2-364
SURFACE4_INFO	MMReg:0xB4C		2-368

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
SURFACE5_LOWER_BOUND	MMReg:0xB54		2-362
SURFACE5_UPPER_BOUND	MMReg:0xB58		2-364
SURFACE5_INFO	MMReg:0xB5C		2-369
SURFACE6_LOWER_BOUND	MMReg:0xB64		2-362
SURFACE6_UPPER_BOUND	MMReg:0xB68		2-364
SURFACE6_INFO	MMReg:0xB6C		2-370
SURFACE7_LOWER_BOUND	MMReg:0xB74		2-362
SURFACE7_UPPER_BOUND	MMReg:0xB78		2-364
SURFACE7_INFO	MMReg:0xB7C		2-370
SURFACE_ACCESS_FLAGS	MMReg:0xBF8		2-357
SURFACE_ACCESS_CLR	MMReg:0xBFC		2-359
DISP_MISC_CNTL	MMReg:0xD00		2-144
DAC_MACRO_CNTL	MMReg:0xD04		2-159
DISP_PWR_MAN	MMReg:0xD08		2-147
DISP_TEST_DEBUG_CNTL	MMReg:0xD10		2-148
DISP_HW_DEBUG	MMReg:0xD14		2-149
DAC_CRC_SIG1	MMReg:0xD18		2-155
DAC_CRC_SIG2	MMReg:0xD1C		2-156
OV0_LIN_TRANS_A	MMReg:0xD20		2-249
OV0_LIN_TRANS_B	MMReg:0xD24		2-249
OV0_LIN_TRANS_C	MMReg:0xD28		2-249
OV0_LIN_TRANS_D	MMReg:0xD2C		2-249
OV0_LIN_TRANS_E	MMReg:0xD30		2-250
OV0_LIN_TRANS_F	MMReg:0xD34		2-250
OV0_GAMMA_0_F	MMReg:0xD40		2-250
OV0_GAMMA_10_1F	MMReg:0xD44		2-250
OV0_GAMMA_20_3F	MMReg:0xD48		2-251
OV0_GAMMA_40_7F	MMReg:0xD4C		2-251
OV0_GAMMA_380_3BF	MMReg:0xD50		2-251
OV0_GAMMA_3C0_3FF	MMReg:0xD54		2-252
DISP_MERGE_CNTL	MMReg:0xD60		2-143
DISP_OUTPUT_CNTL	MMReg:0xD64		2-142
DISP2_MERGE_CNTL	MMReg:0xD68		2-144
DAC_CRC2_SIG1	MMReg:0xD70		2-156
DAC_CRC2_SIG2	MMReg:0xD74		2-157
RMX_HORZ_PHASE	MMReg:0xDBC		2-304
RBBM_STATUS	MMReg:0xE40	MMReg:0x1740	2-79

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
NQWAIT_UNTIL	MMReg:0xE50		2-84
RBBM_DEBUG	MMReg:0xE6C		2-84
RBBM_CMDFIFO_ADDR	MMReg:0xE70		2-81
RBBM_CMDFIFO_DATAL	MMReg:0xE74		2-82
RBBM_CMDFIFO_DATAH	MMReg:0xE78		2-82
RBBM_CMDFIFO_STAT	MMReg:0xE7C		2-82
VENDOR_ID	MMReg:0xF00	pciConfig:0x0	2-2
DEVICE_ID	MMReg:0xF02	pciConfig:0x2	2-2
COMMAND	MMReg:0xF04:R	pciConfig:0x4	2-2
STATUS	MMReg:0xF06:R	pciConfig:0x6	2-3
REVISION_ID	MMReg:0xF08	pciConfig:0x8	2-3
REGPROG_INF	MMReg:0xF09	pciConfig:0x9	2-8
SUB_CLASS	MMReg:0xF0A	pciConfig:0xA	2-5
BASE_CODE	MMReg:0xF0B	pciConfig:0xB	2-4
CACHE_LINE	MMReg:0xF0C:R	pciConfig:0xC	2-8
LATENCY	MMReg:0xF0D:R	pciConfig:0xD	2-8
HEADER	MMReg:0xF0E	pciConfig:0xE	2-7
BIST	MMReg:0xF0F	pciConfig:0xF	2-5
MEM_BASE	MMReg:0xF10:R	pciConfig:0x10	2-4
IO_BASE	MMReg:0xF14:R	pciConfig:0x14	2-3
REG_BASE	MMReg:0xF18:R	pciConfig:0x18	2-4
ADAPTER_ID	MMReg:0xF2C	pciConfig:0x2C	2-5
BIOS_ROM	MMReg:0xF30:R	pciConfig:0x30	2-5
CAPABILITIES_PTR	MMReg:0xF34	pciConfig:0x34	2-6
INTERRUPT_LINE	MMReg:0xF3C:R	pciConfig:0x3C	2-7
INTERRUPT_PIN	MMReg:0xF3D	pciConfig:0x3D	2-8
MIN_GRANT	MMReg:0xF3E	pciConfig:0x3E	2-9
MAX_LATENCY	MMReg:0xF3F	pciConfig:0x3F	2-8
ADAPTER_ID_W	MMReg:0xF4C:R	pciConfig:0x4C	2-4
PMI_CAP_ID	MMReg:0xF50	pciConfig:0x50	2-28
PMI_NXT_CAP_PTR	MMReg:0xF51	pciConfig:0x51	2-28
PMI_PMC_REG	MMReg:0xF52	pciConfig:0x52	2-28
PM_STATUS	MMReg:0xF54:R	pciConfig:0x54	2-28
PMI_DATA	MMReg:0xF57	pciConfig:0x57	2-29
AGP_CAP_ID	MMReg:0xF58	pciConfig:0x58	2-14
AGP_STATUS	MMReg:0xF5C	pciConfig:0x5C	2-14

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Address	Page
AGP_COMMAND	*MMReg:0xF60:R*	*pciConfig:0x60*	*2-13*

A.4 Subpicture Registers Sorted by Name

Table A-4 Subpicture Registers Sorted by Name

Register Name	Address	Page
SUBPIC_0_PAL	*SUBPICIND:0x0*	*2-269*
SUBPIC_1_PAL	*SUBPICIND:0x1*	*2-269*
SUBPIC_2_PAL	*SUBPICIND:0x2*	*2-269*
SUBPIC_3_PAL	*SUBPICIND:0x3*	*2-269*
SUBPIC_4_PAL	*SUBPICIND:0x4*	*2-270*
SUBPIC_5_PAL	*SUBPICIND:0x5*	*2-270*
SUBPIC_6_PAL	*SUBPICIND:0x6*	*2-270*
SUBPIC_7_PAL	*SUBPICIND:0x7*	*2-270*
SUBPIC_8_PAL	*SUBPICIND:0x8*	*2-271*
SUBPIC_9_PAL	*SUBPICIND:0x9*	*2-271*
SUBPIC_A_PAL	*SUBPICIND:0xA*	*2-271*
SUBPIC_B_PAL	*SUBPICIND:0xB*	*2-271*
SUBPIC_C_PAL	*SUBPICIND:0xC*	*2-272*
SUBPIC_D_PAL	*SUBPICIND:0xD*	*2-272*
SUBPIC_E_PAL	*SUBPICIND:0xE*	*2-272*
SUBPIC_F_PAL	*SUBPICIND:0xF*	*2-272*

A.5 Clock Index Registers Sorted by Name

Table A-5 Clock Index Registers Sorted by Name

Register Name	Address	Page
AGP_PLL_CNTL	CLKIND:0xB	2-38
CG_TEST_MACRO_RW_CNTL	CLKIND:0x19	2-56
CG_TEST_MACRO_RW_DATA	CLKIND:0x18	2-56
CG_TEST_MACRO_RW_READ	CLKIND:0x17	2-56
CG_TEST_MACRO_RW_WRITE	CLKIND:0x16	2-56
CLK_PIN_CNTL	CLKIND:0x1	2-33
CLK_PWRMGT_CNTL	CLKIND:0x14	2-30
CLK_PWRMGT_CNTL	CLKIND:0x14	2-49
DISP_TEST_MACRO_RW_CNTL	CLKIND:0x1D	2-301
DISP_TEST_MACRO_RW_DATA	CLKIND:0x1C	2-301
DISP_TEST_MACRO_RW_READ	CLKIND:0x1B	2-301
DISP_TEST_MACRO_RW_WRITE	CLKIND:0x1A	2-301
HTOTAL_CNTL	CLKIND:0x9	2-45
HTOTAL_CNTL	CLKIND:0x9	2-146
HTOTAL2_CNTL	CLKIND:0x2E	2-48
HTOTAL2_CNTL	CLKIND:0x2E	2-146
M_SPLL_REF_FB_DIV	CLKIND:0xA	2-46
M_SPLL_REF_FB_DIV	CLKIND:0xA	2-57
MCLK_CNTL	CLKIND:0x12	2-54
MCLK_MISC	CLKIND:0x1F	2-55
MDLL_CKO	CLKIND:0xF	2-53
MDLL_RDCKA	CLKIND:0x10	2-53
MPLL_CNTL	CLKIND:0xE	2-52
P2PLL_CNTL	CLKIND:0x2A	2-41
P2PLL_DIV_0	CLKIND:0x2C	2-47
P2PLL_DIV_0	CLKIND:0x2C	2-65
P2PLL_DIV_0	CLKIND:0x2C	2-280
P2PLL_REF_DIV	CLKIND:0x2B	2-41
PIXCLKS_CNTL	CLKIND:0x2D	2-63
PIXCLKS_CNTL	CLKIND:0x2D	2-278
PLL_PWRMGT_CNTL	CLKIND:0x15	2-31
PLL_PWRMGT_CNTL	CLKIND:0x15	2-50
PLL_TEST_CNTL	CLKIND:0x13	2-46
PLL_TEST_CNTL	CLKIND:0x13	2-58
PLL_TEST_CNTL	CLKIND:0x13	2-64

Table A-5 Clock Index Registers Sorted by Name (Continued)

Register Name	Address	Page
PLL_TEST_CNTL	CLKIND:0x13	2-279
PPLL_CNTL	CLKIND:0x2	2-34
PPLL_DIV_0	CLKIND:0x4	2-42
PPLL_DIV_0	CLKIND:0x4	2-60
PPLL_DIV_0	CLKIND:0x4	2-275
PPLL_DIV_1	CLKIND:0x5	2-43
PPLL_DIV_1	CLKIND:0x5	2-60
PPLL_DIV_1	CLKIND:0x5	2-275
PPLL_DIV_2	CLKIND:0x6	2-44
PPLL_DIV_2	CLKIND:0x6	2-61
PPLL_DIV_2	CLKIND:0x6	2-276
PPLL_DIV_3	CLKIND:0x7	2-44
PPLL_DIV_3	CLKIND:0x7	2-62
PPLL_DIV_3	CLKIND:0x7	2-277
PPLL_REF_DIV	CLKIND:0x3	2-35
SCLK_CNTL	CLKIND:0xD	2-36
SPLL_CNTL	CLKIND:0xC	2-35
TV_DTO_INCREMENTS	CLKIND:0x23	2-40
TV_PLL_CNTL	CLKIND:0x21	2-39
TV_PLL_CNTL1	CLKIND:0x22	2-40
TV_PLL_FINE_CNTL	CLKIND:0x20	2-39
VCLK_ECP_CNTL	CLKIND:0x8	2-62
VCLK_ECP_CNTL	CLKIND:0x8	2-277

A.6 I/O Registers Sorted by Name

Table A-6 I/O Registers Sorted by Name

Register Name	I/O Address	Memory Address	Secondary Address	Page
BIOS_0_SCRATCH	IOReg:0x10	MMReg:0x10		2-390
BIOS_1_SCRATCH	IOReg:0x14	MMReg:0x14		2-390
BIOS_2_SCRATCH	IOReg:0x18	MMReg:0x18		2-390
BIOS_3_SCRATCH	IOReg:0x1C	MMReg:0x1C		2-390
BIOS_4_SCRATCH	IOReg:0x20	MMReg:0x20		2-390
BIOS_5_SCRATCH	IOReg:0x24	MMReg:0x24		2-391
BIOS_6_SCRATCH	IOReg:0x28	MMReg:0x28		2-391
BIOS_7_SCRATCH	IOReg:0x2C	MMReg:0x2C		2-391
BUS_CNTL	IOReg:0x30	MMReg:0x30		2-10
BUS_CNTL1	IOReg:0x34	MMReg:0x34		2-11
CLOCK_CNTL_DATA	IOReg:0xC	MMReg:0xC		2-42
CLOCK_CNTL_DATA	IOReg:0xC	MMReg:0xC		2-57
CLOCK_CNTL_DATA	IOReg:0xC	MMReg:0xC		2-59
CLOCK_CNTL_DATA	IOReg:0xC	MMReg:0xC		2-274
CLOCK_CNTL_INDEX	IOReg:0x8	MMReg:0x8		2-42
CLOCK_CNTL_INDEX	IOReg:0x8	MMReg:0x8		2-57
CLOCK_CNTL_INDEX	IOReg:0x8	MMReg:0x8		2-59
CLOCK_CNTL_INDEX	IOReg:0x8	MMReg:0x8		2-274
CONFIG_CNTL	IOReg:0xE0	MMReg:0xE0		2-6
CONFIG_MEMSIZE	IOReg:0xF8	MMReg:0xF8		2-6
CONFIG_XSTRAP	IOReg:0xE4	MMReg:0xE4		2-387
CRTC_EXT_CNTL	IOReg:0x54	MMReg:0x54		2-212
CRTC_EXT_CNTL	IOReg:0x54	MMReg:0x54		2-341
CRTC_GEN_CNTL	IOReg:0x50	MMReg:0x50		2-21
CRTC_GEN_CNTL	IOReg:0x50	MMReg:0x50		2-209
CRTC_STATUS	IOReg:0x5C	MMReg:0x5C		2-223
DAC_CNTL	IOReg:0x58	MMReg:0x58		2-18
DAC_CNTL	IOReg:0x58	MMReg:0x58		2-151
DAC_CNTL2	IOReg:0x7C	MMReg:0x7C		2-157
GEN_INT_CNTL	IOReg:0x40	MMReg:0x40		2-84
GEN_INT_CNTL	IOReg:0x40	MMReg:0x40		2-137
GEN_INT_CNTL	IOReg:0x40	MMReg:0x40		2-165
GEN_INT_CNTL	IOReg:0x40	MMReg:0x40		2-330
GEN_INT_CNTL	IOReg:0x40	MMReg:0x40		2-337
GEN_INT_CNTL	IOReg:0x40	MMReg:0x40		2-384
GEN_INT_STATUS	IOReg:0x44	MMReg:0x44		2-86
GEN_INT_STATUS	IOReg:0x44	MMReg:0x44		2-139
GEN_INT_STATUS	IOReg:0x44	MMReg:0x44		2-166
GEN_INT_STATUS	IOReg:0x44	MMReg:0x44		2-332

Table A-6 I/O Registers Sorted by Name (Continued)

Register Name	I/O Address	Memory Address	Secondary Address	Page
GEN_INT_STATUS	IOReg:0x44	MMReg:0x44		2-338
GEN_INT_STATUS	IOReg:0x44	MMReg:0x44		2-372
GPIO_CRT2_DDC	IOReg:0x6C	MMReg:0x6C		2-226
GPIO_DVI_DDC	IOReg:0x64	MMReg:0x64		2-224
GPIO_MONID	IOReg:0x68	MMReg:0x68		2-225
GPIO_VGA_DDC	IOReg:0x60	MMReg:0x60		2-224
HI_STAT	IOReg:0x4C	MMReg:0x4C		2-12
MEM_VGA_RP_SEL	IOReg:0x3C	MMReg:0x3C		2-336
MEM_VGA_WP_SEL	IOReg:0x38	MMReg:0x38		2-336
MM_DATA	IOReg:0x4	MMReg:0x4		2-15
MM_INDEX	IOReg:0x0	MMReg:0x0		2-14
PALETTE_30_DATA	IOReg:0xB8	MMReg:0xB8		2-303
PALETTE_DATA	IOReg:0xB4	MMReg:0xB4		2-303
PALETTE_INDEX	IOReg:0xB0	MMReg:0xB0		2-303
RBBM_CNTL	IOReg:0xEC	MMReg:0xEC	MMReg:0xE44	2-77
RBBM_SOFT_RESET	IOReg:0xF0	MMReg:0xF0	MMReg:0xE48	2-78

A.7 VGA Registers Sorted by Name

Table A-7 VGA Registers Sorted by Name

Register Name	Address		Page
ATTR00	VGAATTRIND:0x0		2-193
ATTR01	VGAATTRIND:0x1		2-193
ATTR02	VGAATTRIND:0x2		2-193
ATTR03	VGAATTRIND:0x3		2-193
ATTR04	VGAATTRIND:0x4		2-194
ATTR05	VGAATTRIND:0x5		2-194
ATTR06	VGAATTRIND:0x6		2-194
ATTR07	VGAATTRIND:0x7		2-194
ATTR08	VGAATTRIND:0x8		2-195
ATTR09	VGAATTRIND:0x9		2-195
ATTR0A	VGAATTRIND:0xA		2-195
ATTR0B	VGAATTRIND:0xB		2-195
ATTR0C	VGAATTRIND:0xC		2-196
ATTR0D	VGAATTRIND:0xD		2-196
ATTR0E	VGAATTRIND:0xE		2-196
ATTR0F	VGAATTRIND:0xF		2-196
ATTR10	VGAATTRIND:0x10		2-197
ATTR11	VGAATTRIND:0x11		2-197
ATTR12	VGAATTRIND:0x12		2-198
ATTR13	VGAATTRIND:0x13		2-198
ATTR14	VGAATTRIND:0x14		2-199
ATTRDR	VGA_IO:0x3C1		2-192
ATTRDW	VGA_IO:0x3C0		2-192
ATTRX	VGA_IO:0x3C0		2-192
CRT00	VGACRTIND:0x0		2-172
CRT00_S	VGACRTIND:0x40		2-181
CRT01	VGACRTIND:0x1		2-172
CRT01_S	VGACRTIND:0x41		2-182
CRT02	VGACRTIND:0x2		2-173
CRT02_S	VGACRTIND:0x42		2-182
CRT03	VGACRTIND:0x3		2-173
CRT03_S	VGACRTIND:0x43		2-182
CRT04	VGACRTIND:0x4		2-173
CRT04_S	VGACRTIND:0x44		2-182
CRT05	VGACRTIND:0x5		2-174
CRT05_S	VGACRTIND:0x45		2-183
CRT06	VGACRTIND:0x6		2-174
CRT06_S	VGACRTIND:0x46		2-183
CRT07	VGACRTIND:0x7		2-174

Table A-7 VGA Registers Sorted by Name (Continued)

Register Name	Address		Page
CRT07_S	VGACRTIND:0x47		2-183
CRT08	VGACRTIND:0x8		2-175
CRT08_S	VGACRTIND:0x48		2-184
CRT09	VGACRTIND:0x9		2-175
CRT09_S	VGACRTIND:0x49		2-184
CRT0A	VGACRTIND:0xA		2-176
CRT0A_S	VGACRTIND:0x4A		2-185
CRT0B	VGACRTIND:0xB		2-176
CRT0B_S	VGACRTIND:0x4B		2-185
CRT0C	VGACRTIND:0xC		2-177
CRT0C_S	VGACRTIND:0x4C		2-185
CRT0D	VGACRTIND:0xD		2-177
CRT0D_S	VGACRTIND:0x4D		2-186
CRT0E	VGACRTIND:0xE		2-177
CRT0E_S	VGACRTIND:0x4E		2-186
CRT0F	VGACRTIND:0xF		2-178
CRT0F_S	VGACRTIND:0x4F		2-186
CRT10	VGACRTIND:0x10		2-178
CRT10_S	VGACRTIND:0x50		2-186
CRT11	VGACRTIND:0x11		2-178
CRT11_S	VGACRTIND:0x51		2-187
CRT12	VGACRTIND:0x12		2-179
CRT12_S	VGACRTIND:0x52		2-187
CRT13	VGACRTIND:0x13		2-179
CRT13_S	VGACRTIND:0x53		2-187
CRT14	VGACRTIND:0x14		2-179
CRT14	VGACRTIND:0x14		2-352
CRT14_S	VGACRTIND:0x54		2-188
CRT14_S	VGACRTIND:0x54		2-354
CRT15	VGACRTIND:0x15		2-180
CRT15_S	VGACRTIND:0x55		2-188
CRT16	VGACRTIND:0x16		2-180
CRT16_S	VGACRTIND:0x56		2-188
CRT17	VGACRTIND:0x17		2-180
CRT17	VGACRTIND:0x17		2-353
CRT17_S	VGACRTIND:0x57		2-188
CRT17_S	VGACRTIND:0x57		2-354
CRT18	VGACRTIND:0x18		2-181
CRT18_S	VGACRTIND:0x58		2-189
CRT1E	VGACRTIND:0x1E		2-353
CRT1E_S	VGACRTIND:0x5E		2-355

Table A-7 VGA Registers Sorted by Name (Continued)

Register Name	Address		Page
CRT1F	VGACRTIND:0x1F		2-353
CRT1F_S	VGACRTIND:0x5F		2-355
CRT22	VGACRTIND:0x22		2-354
CRT22_S	VGACRTIND:0x62		2-356
CRTC8_DATA	VGA_IO:0x3B5	VGA_IO:0x3D5	2-172
CRTC8_DATA	VGA_IO:0x3B5	VGA_IO:0x3D5	2-352
CRTC8_IDX	VGA_IO:0x3B4	VGA_IO:0x3D4	2-172
CRTC8_IDX	VGA_IO:0x3B4	VGA_IO:0x3D4	2-352
DAC_DATA	VGA_IO:0x3C9		2-154
DAC_MASK	VGA_IO:0x3C6		2-154
DAC_R_INDEX	VGA_IO:0x3C7		2-154
DAC_W_INDEX	VGA_IO:0x3C8		2-154
GENENB	VGA_IO:0x3C3		2-16
GENFC_RD	VGA_IO:0x3CA		2-160
GENFC_WT	VGA_IO:0x3BA	VGA_IO:0x3DA	2-160
GENMO_RD	VGA_IO:0x3CC		2-17
GENMO_RD	VGA_IO:0x3CC		2-161
GENMO_RD	VGA_IO:0x3CC		2-343
GENMO_WT	VGA_IO:0x3C2		2-16
GENMO_WT	VGA_IO:0x3C2		2-160
GENMO_WT	VGA_IO:0x3C2		2-344
GENS0	VGA_IO:0x3C2		2-162
GENS1	VGA_IO:0x3BA	VGA_IO:0x3DA	2-162
GRA00	VGAGRPHIND:0x0		2-346
GRA01	VGAGRPHIND:0x1		2-346
GRA02	VGAGRPHIND:0x2		2-347
GRA03	VGAGRPHIND:0x3		2-347
GRA04	VGAGRPHIND:0x4		2-348
GRA05	VGAGRPHIND:0x5		2-190
GRA05	VGAGRPHIND:0x5		2-348
GRA06	VGAGRPHIND:0x6		2-349
GRA07	VGAGRPHIND:0x7		2-349
GRA08	VGAGRPHIND:0x8		2-349
GRPH8_DATA	VGA_IO:0x3CF		2-190
GRPH8_DATA	VGA_IO:0x3CF		2-346
GRPH8_IDX	VGA_IO:0x3CE		2-190
GRPH8_IDX	VGA_IO:0x3CE		2-346
SEQ00	VGASEQIND:0x0		2-170
SEQ01	VGASEQIND:0x1		2-170
SEQ02	VGASEQIND:0x2		2-350
SEQ03	VGASEQIND:0x3		2-171
SEQ04	VGASEQIND:0x4		2-351

Table A-7 VGA Registers Sorted by Name (Continued)

Register Name	Address		Page
SEQ8_DATA	*VGA_IO:0x3C5*		*2-171*
SEQ8_DATA	*VGA_IO:0x3C5*		*2-350*
SEQ8_IDX	*VGA_IO:0x3C4*		*2-171*
SEQ8_IDX	*VGA_IO:0x3C4*		*2-350*

A.8 VGA Registers Sorted by Address

Table A-8 VGA Registers Sorted by Address

Register Name	Address		Page
CRTC8_IDX	VGA_IO:0x3B4	VGA_IO:0x3D4	2-172
CRTC8_IDX	VGA_IO:0x3B4	VGA_IO:0x3D4	2-352
CRTC8_DATA	VGA_IO:0x3B5	VGA_IO:0x3D5	2-172
CRTC8_DATA	VGA_IO:0x3B5	VGA_IO:0x3D5	2-352
GENFC_WT	VGA_IO:0x3BA	VGA_IO:0x3DA	2-160
GENS1	VGA_IO:0x3BA	VGA_IO:0x3DA	2-162
ATTRDW	VGA_IO:0x3C0		2-192
ATTRX	VGA_IO:0x3C0		2-192
ATTRDR	VGA_IO:0x3C1		2-192
GENMO_WT	VGA_IO:0x3C2		2-16
GENMO_WT	VGA_IO:0x3C2		2-160
GENMO_WT	VGA_IO:0x3C2		2-344
GENS0	VGA_IO:0x3C2		2-162
GENENB	VGA_IO:0x3C3		2-16
SEQ8_IDX	VGA_IO:0x3C4		2-171
SEQ8_IDX	VGA_IO:0x3C4		2-350
SEQ8_DATA	VGA_IO:0x3C5		2-171
SEQ8_DATA	VGA_IO:0x3C5		2-350
DAC_MASK	VGA_IO:0x3C6		2-154
DAC_R_INDEX	VGA_IO:0x3C7		2-154
DAC_W_INDEX	VGA_IO:0x3C8		2-154
DAC_DATA	VGA_IO:0x3C9		2-154
GENFC_RD	VGA_IO:0x3CA		2-160
GENMO_RD	VGA_IO:0x3CC		2-17
GENMO_RD	VGA_IO:0x3CC		2-161
GENMO_RD	VGA_IO:0x3CC		2-343
GRPH8_IDX	VGA_IO:0x3CE		2-190
GRPH8_IDX	VGA_IO:0x3CE		2-346
GRPH8_DATA	VGA_IO:0x3CF		2-190
GRPH8_DATA	VGA_IO:0x3CF		2-346
ATTR00	VGAATTRIND:0x0		2-193
ATTR01	VGAATTRIND:0x1		2-193
ATTR10	VGAATTRIND:0x10		2-197
ATTR11	VGAATTRIND:0x11		2-197
ATTR12	VGAATTRIND:0x12		2-198
ATTR13	VGAATTRIND:0x13		2-198
ATTR14	VGAATTRIND:0x14		2-199
ATTR02	VGAATTRIND:0x2		2-193

Table A-8 VGA Registers Sorted by Address (Continued)

Register Name	Address		Page
ATTR03	VGAATTRIND:0x3		2-193
ATTR04	VGAATTRIND:0x4		2-194
ATTR05	VGAATTRIND:0x5		2-194
ATTR06	VGAATTRIND:0x6		2-194
ATTR07	VGAATTRIND:0x7		2-194
ATTR08	VGAATTRIND:0x8		2-195
ATTR09	VGAATTRIND:0x9		2-195
ATTR0A	VGAATTRIND:0xA		2-195
ATTR0B	VGAATTRIND:0xB		2-195
ATTR0C	VGAATTRIND:0xC		2-196
ATTR0D	VGAATTRIND:0xD		2-196
ATTR0E	VGAATTRIND:0xE		2-196
ATTR0F	VGAATTRIND:0xF		2-196
CRT00	VGACRTIND:0x0		2-172
CRT01	VGACRTIND:0x1		2-172
CRT10	VGACRTIND:0x10		2-178
CRT11	VGACRTIND:0x11		2-178
CRT12	VGACRTIND:0x12		2-179
CRT13	VGACRTIND:0x13		2-179
CRT14	VGACRTIND:0x14		2-179
CRT14	VGACRTIND:0x14		2-352
CRT15	VGACRTIND:0x15		2-180
CRT16	VGACRTIND:0x16		2-180
CRT17	VGACRTIND:0x17		2-180
CRT17	VGACRTIND:0x17		2-353
CRT18	VGACRTIND:0x18		2-181
CRT1E	VGACRTIND:0x1E		2-353
CRT1F	VGACRTIND:0x1F		2-353
CRT02	VGACRTIND:0x2		2-173
CRT22	VGACRTIND:0x22		2-354
CRT03	VGACRTIND:0x3		2-173
CRT04	VGACRTIND:0x4		2-173
CRT00_S	VGACRTIND:0x40		2-181
CRT01_S	VGACRTIND:0x41		2-182
CRT02_S	VGACRTIND:0x42		2-182
CRT03_S	VGACRTIND:0x43		2-182
CRT04_S	VGACRTIND:0x44		2-182
CRT05_S	VGACRTIND:0x45		2-183
CRT06_S	VGACRTIND:0x46		2-183
CRT07_S	VGACRTIND:0x47		2-183
CRT08_S	VGACRTIND:0x48		2-184

Table A-8 VGA Registers Sorted by Address (Continued)

Register Name	Address		Page
CRT09_S	VGACRTIND:0x49		2-184
CRT0A_S	VGACRTIND:0x4A		2-185
CRT0B_S	VGACRTIND:0x4B		2-185
CRT0C_S	VGACRTIND:0x4C		2-185
CRT0D_S	VGACRTIND:0x4D		2-186
CRT0E_S	VGACRTIND:0x4E		2-186
CRT0F_S	VGACRTIND:0x4F		2-186
CRT05	VGACRTIND:0x5		2-174
CRT10_S	VGACRTIND:0x50		2-186
CRT11_S	VGACRTIND:0x51		2-187
CRT12_S	VGACRTIND:0x52		2-187
CRT13_S	VGACRTIND:0x53		2-187
CRT14_S	VGACRTIND:0x54		2-188
CRT14_S	VGACRTIND:0x54		2-354
CRT15_S	VGACRTIND:0x55		2-188
CRT16_S	VGACRTIND:0x56		2-188
CRT17_S	VGACRTIND:0x57		2-188
CRT17_S	VGACRTIND:0x57		2-354
CRT18_S	VGACRTIND:0x58		2-189
CRT1E_S	VGACRTIND:0x5E		2-355
CRT1F_S	VGACRTIND:0x5F		2-355
CRT06	VGACRTIND:0x6		2-174
CRT22_S	VGACRTIND:0x62		2-356
CRT07	VGACRTIND:0x7		2-174
CRT08	VGACRTIND:0x8		2-175
CRT09	VGACRTIND:0x9		2-175
CRT0A	VGACRTIND:0xA		2-176
CRT0B	VGACRTIND:0xB		2-176
CRT0C	VGACRTIND:0xC		2-177
CRT0D	VGACRTIND:0xD		2-177
CRT0E	VGACRTIND:0xE		2-177
CRT0F	VGACRTIND:0xF		2-178
GRA00	VGAGRPHIND:0x0		2-346
GRA01	VGAGRPHIND:0x1		2-346
GRA02	VGAGRPHIND:0x2		2-347
GRA03	VGAGRPHIND:0x3		2-347
GRA04	VGAGRPHIND:0x4		2-348
GRA05	VGAGRPHIND:0x5		2-190
GRA05	VGAGRPHIND:0x5		2-348
GRA06	VGAGRPHIND:0x6		2-349
GRA07	VGAGRPHIND:0x7		2-349
GRA08	VGAGRPHIND:0x8		2-349

Table A-8 VGA Registers Sorted by Address (Continued)

Register Name	Address		Page
SEQ00	*VGASEQIND:0x0*		*2-170*
SEQ01	*VGASEQIND:0x1*		*2-170*
SEQ02	*VGASEQIND:0x2*		*2-350*
SEQ03	*VGASEQIND:0x3*		*2-171*
SEQ04	*VGASEQIND:0x4*		*2-351*

This page intentionally left blank.

Appendix B
Revision History

B.1 Rev 0.01, August 2000

Based on engineering document version 1.60.

This page intentionally left blank.