RAGE™128 PRO
Register Reference Guide

Technical Reference Manual

P/N: RRG-G04500-C Rev 1.01

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
0.01	Jan. 1999
0.02	Feb. 1999
0.03	Mar. & Apr. 1999
0.04	May 1999
0.05	June 1999
1.01	Jan 2000
For more details, refer to Appendix B.	

Related Manuals

RAGE 128 series

- RAGE™ 128 Register Reference Guide (RRG-R04100)
- RAGE™ 128 Graphics Controller Specifications (GCS-C04100)

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 RAGE 128 PRO 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, VGA, etc., which will be indicated on the appropriate registers).

1.2.5 Typical Register Format and Explanation

Table 1-1 Typical Register Format

VENDOR_ID CFG:0x0000 MMR:0x0F00 MMR_1:0x0F00 IND:0x0F00 [R] 16 bits (access: 8/16/32)		
Field Name	**Bit(s)**	**Description**
CRTC_V_TOTAL	10:0	Vertical total
(Reserved)	15:11	
CRTC_V_DISP	26:16	Vertical display end
(Reserved)	31:27	

Table 1-2 Explanation of Format

Register Variable	Example
Register Name	VENDOR_ID
Read / Write Capability R = Readable W = Writable	[R]
Register Size	16 bits
Accessibility by Bit Size	(access: 8/16/32)
Register Addresses	MMR:0x0F00 MMR_1:0x0F00 IND:0x0F00
Field Name	CRTC_V_TOTAL
Field position/size	26:16
Field description	Vertical total

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 1-3 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
Overview and Memory Mapping

2.1 General Classification

For ease of discussion and reference, the registers are grouped into the following main classes according to their functionality:

- Setup and Configuration registers
- Host Interface (PCI Configuration Space and AGP) registers
- VGA registers
- Accelerator CRTC and DAC registers
- 2D Engine registers
- Front-end Scaling and 3D operations registers
- Concurrent Command Engine registers
- Multimedia registers
- Miscellaneous registers

Note that these are general register classes only. There are instances when specific bit fields of the same register may belong to a different class register. In such cases, it is noted in the register description.

The following is an overview of all the registers. As can be seen, some classes are further divided into sub-groups.

2.1.1 Setup and Configuration Registers

Setup and configuration registers are memory mapped and aliased at an I/O address. Most of these registers are initialized only once at boot time. They are further divided into:

- **General I/O Control register** — used to configure the General Purpose I/O pins on the accelerator chip.
- **Bus Control register** — used to configure the on-chip bus interface unit.
- **Memory registers** — used to configure the memory interfaces.
- **Test and Debug registers** — used for chip diagnostics and hardware debugging.
- **Configuration registers** — used to configure the memory aperture and to read the current board configuration.

2.1.2 Host Interface Registers

- **PCI Configuration Space Registers** — used to determine the host bus configuration during system reset. For the RAGE 128, the internal host bus interface has been optimized to support the PCI Version 2.1 bus configuration, providing full 32-bit memory and I/O operations.
- **AGP Registers** — used to configure the Accelerated Graphic Port.

2.1.3 VGA Registers

The VGA registers provide register-level compatibility with the IBM VGA display adapter. They and the accelerator registers are completely segregated from each other, and their functions are mutually exclusive.

2.1.4 Accelerator CRTC and DAC Registers

Accelerator CRTC and DAC Registers are memory mapped and aliased at an I/O address. (Note that accelerator CRTC registers are not the same as the VGA CRTC registers.) They are further divided into the following groups:

- **Accelerator CRTC registers** — used to configure the CRT controller.
- **Clock Control register** — used to configure the pixel clock.
- **PLL registers** — accessed indirectly through the Clock Control register.
- **DAC Control registers** — used to configure the DAC.
- **Overscan registers** — used to configure overscan borders.
- **Hardware Cursor registers** — used to define and move the hardware cursor.

2.1.5 2D Engine Registers

These are divided into two main groups: Trajectory registers and Draw Engine Control registers:

Trajectory Registers

Trajectory registers are memory mapped. They set up the source and destination trajectories and initiate draw operations. They are further divided into two groups:

- **Destination Trajectory registers** — used to define the region in which pixels are drawn. The region may be a a line, a rectangular, or a trapezoidal area.
- **Source Trajectory registers** — used to define a rectangular region from which pixel data is taken. The pixel data may be used as a monochrome or color pixel source, or a polygon fill mask.

Draw Engine Control Registers

Draw Engine Control Registers are memory mapped. They set up the source pixel data, the draw engine data path, and the destination mixing logic. They are divided into the following groups:

- **Host Data registers** — used for transferring data from the host to the draw engine.
- **Pattern registers** — used to enable and define fixed patterns.
- **Scissor registers** — used to define a draw region.
- **Data Path registers** — used to configure the data path and ALU.
- **Color Compare registers** — used to configure the source and destination color compare.
- **Draw Engine Composite Control register** — abbreviated composites of other draw engine control registers.
- **Draw Engine Status register** — used to report the current state of the draw engine.

2.1.6 3D Engine Registers

The 3D Engine registers are memory mapped and are further divided into the following groups:

- **Front-End Scaler Pipe registers** — used to configure the front-end scaler source data and to control any subsequent blending, color conversion, and dithering. Most of the scaler registers are aliased with certain 3D and Texture Mapping registers.
- **Texture Mapping registers** — used to hold the 'S' and 'T' sample address offsets to the start of the available mipmaps, and to configure the associated quadratic interpolators.
- **Specular, Color, Z, and Alpha Interpolator registers** — used to configure the specular interpolation, the Z buffering and interpolation, the RGB and alpha interpolation, alpha blending, and fogging.
- **Setup Engine registers** — used to setup the draw and color/texture functions.

2.1.7 Concurrent Command Engine Registers

- **Vertex Controller and Floating Point Unit Registers** — used to configure the vertex triangle walker and the floating point area calculation.
- **Status/Data/Address registers** — used to obtain status of command engine.

2.1.8 Multimedia Registers

These are registers used for multimedia operations such as video capture and playback. They are divided into the following groups:

- **Overlay Window registers** — used to specify the overlayed scaling window dimensions and coordinates to be displayed.
- **Overlay Scaler registers** — used to set up the scaling factors.
- **Video Capture registers** — used to initialize, set the video configuration, define the capture buffer requirements, and trigger the capture.
- **Multimedia Peripheral Port (MPP) registers** — used to configure and access the MPP.
- **Subpicture registers** — used to control DVD subpicture feature.
- **VIP Port registers** — used to control VIP multimedia port.
- **Hardware Assisted I2C registers** — used to control a 16-entry deep buffer for storing out-going or in-coming data.
- **iDCT registers** — used to control Inverse DCT engine.

2.1.9 Miscellaneous Registers

These are registers that do not quite fit into any of the groups above. Amongst them are:

- **Scratch Pad registers —** used for general purpose storage for the adapter ROM and for communicating the adapter ROM segment location to host applications. In test modes, these registers are used for chip diagnostics.

2.2 Memory Mapping

The RAGE 128 uses a fully memory mapped programming model as shown in the diagram below.

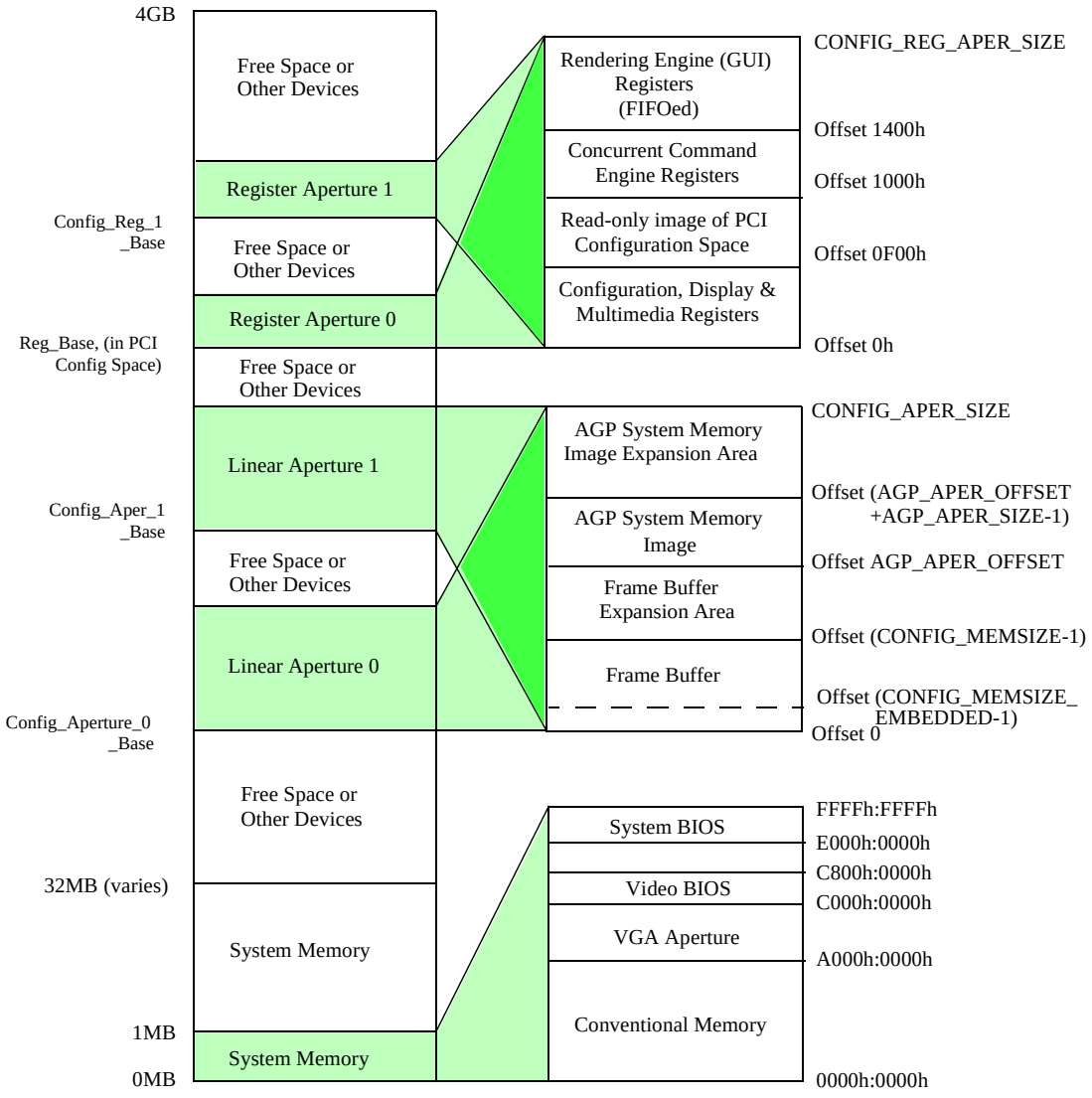

Figure 2-1. Register and Frame Buffer Mapping Within the Host Address Space

2.2.1 Description of Mapped Memory Apertures

Table 2-1 Memory Mapping Summary

Register Group	Space/Offset
PCI POS registers	PCI Configuration Space
VGA compatible registers	Standard VGA addresses in the I/0 space
Non-GUI registers, memory mapped as well as directly accessible in IOR space	0000h - 00FFh
Non-GUI registers, not accessible in IOR space	0100h - 0EFFh
Read-only copy of PCI configuration space	0F00h - 0FFFh
Concurrent Command Engine registers	1000h - 13FFh
GUI registers	1400h - 1FFFh

Video BIOS

The video BIOS for RAGE 128 is re-locatable using the PCI configuration space. The system BIOS will normally shadow the entire BIOS image to the area starting at segment C000h during system initialization. To directly access the actual BIOS memory, as opposed to the shadow copy in system memory, contact ATI for details.

PCI Configuration Space

The PCI POS registers exist only in the PCI configuration space.

VGA Memory Aperture

When enabled for VGA, RAGE 128 claims the standard VGA resources. The VGA memory aperture position and size are determined by the GRPH_ADRSEL bits. For most VGA graphics modes the aperture is 128KB starting at segment A000h.

Register Apertures

RAGE 128 uses two identical copies of the relocatable memory-mapped register aperture. For the PowerMac environment, this allows one aperture to be marked as cacheable. For the Wintel architecture, the second aperture may be used (but this serves no valid purpose).

These apertures contain all the direct-accessed registers on the chip (except VGA and PCI configuration registers). They also have index/data pairs for all indirectly accessed registers and memories.

To determine the base address of Register Aperture 0:

1. Use the REG_BASE register in the PCI configuration space.

-Or-

2. Read from the I/O Register Aperture using MM_INDEX <= F18h and read MM_DATA.

To determine the base address of Register Aperture 1, use the CONFIG_REG_1_BASE register, which can be read in Register Aperture 0 once its base has been found as indicated above. Reading CONFIG_REG_1_BASE is the only method of determining the location of Register Aperture 1 that is forward compatible with future generations of hardware.

To determine the size of each register aperture, use the CONFIG_REG_APER_SIZE register. The size may vary in future generations of the accelerator.

Linear Memory Apertures

There are also two identical copies of the relocatable Linear Memory Aperture in RAGE 128. For the PowerMac environment, this allows each to be independently marked as big-endian or little-endian. For the Wintel architecture, the second aperture may also be used (but this serves no valid purpose).

These apertures allow access to the frame buffer memory, and in AGP systems, access to the AGP memory as seen by the RAGE 128.

To determine the base address of Linear Aperture 0, use the CONFIG_APER_0_BASE register. To determine the base address of Linear Aperture 1, use the CONFIG_APER_1_BASE register. Both these registers can be read in any register aperture.

To determine the size of each linear aperture, use the CONFIG_APER_SIZE register. The size may vary in future generations of the accelerator.

Frame Buffer

The frame buffer image occupies the area in each aperture from offset 0 to CONFIG_MEMSIZE-1.

When CONFIG_MEMSIZE_EMBEDDED is greater than 0, the hardware has on-chip memory for the first piece of the frame buffer. This embedded memory is included in the CONFIG_MEMSIZE total. The RAGE 128 does not have any embedded memory.

The RAGE 128 supports up to 32MB of frame buffer memory. This limit may be expanded for future hardware generations, therefore the software should use the procedures outlined here for determining the aperture base addresses, AGP offsets and aperture size.

AGP System Memory Image

Each Linear Aperture contains an image of the AGP system memory as seen by the accelerator. This image starts at offset AGP_APER_OFFSET in the aperture.

The AGP image is intended for debug work. It allows a method to flush out pending AGP cycles still in the host chipset before the software directly accesses system memory. Software would normally directly access AGP system memory using the system processor. Using this AGP image will generate an AGP slave and an AGP bus master cycle for each access (or group of accesses), and therefore it is not recommended.

To determine the size of the AGP memory, use the AGP_APER_SIZE register. This register is an enumerated type that must be converted into a number (refer to the register definition).

The RAGE 128 supports up to 32MB of AGP memory. This limit may be expanded for future hardware generations, therefore software should use the procedures outlined here for determining the aperture base addresses, AGP offsets and aperture size.

2.2.2 Accessing Bytes, Words, and Dwords

The table below shows the register groups and how they are accessed (bytes, words, or Dwords).

Table 2-2 Accessing Registers

Register Group	Byte Addressing	Word Addressing	Dword Addressing
PCI POS registers	✔	✔	✔
VGA registers	✔	Note 1	Note 1
Display & Configuration	✔	✔ (Note 2)	✔ (Note 2)
GUI registers	✘	✘	✔
Multimedia registers	✘ (Note 3)	✘ (Note 3)	✔
PLL registers	✔	✘	✘

Notes:

1. If two or four VGA registers are continuous in the I/O space, 16 or 32 cycles may be used. The cycle will be broken up internally into 2 or 4 sequential cycles starting with the lowest address first.
2. The DAC_REGS register is actually four 8-bit registers. Word or Dword cycles will be split internally into 2 or 4 sequential cycles starting with the lowest address.
3. The multimedia registers that appear in I/O space are Dword-only registers. This means 32-bit IN or OUT operations must be used.
4. When performing a byte or word access to a 32-bit register, simply add 1, 2 or 3 to the absolute address.
5. It is not recommended to perform word or Dword cycles that span a Dword boundary. This will not work correctly in all cases.

2.2.3 Non-Intel Based Memory Mapping

When incorporating the RAGE 128 into a non-Intel platform (such as the Apple Power Macintosh), make sure the platform conforms to the PCI specification. For information on how to configure the RAGE 128 in non-Intel environments, refer to Chapter 2 of the *mach64 Programmer's Guide*.

This page intentionally left blank.

Chapter 3
Register Description

3.1 Scope

This chapter describes the contents of the registers.

3.2 Configuration Registers

Standard PCI Configuration Registers

VENDOR_ID CFG:0x0000 MMR:0x0F00 MMR_1:0x0F00 IND:0x0F00 [R] 16 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VENDOR_ID	15:0	0x1002	ATI Vendor ID Number.

VENDOR_ID: Vendor ID Description.

DEVICE_ID CFG:0x0002 MMR:0x0F02 MMR_1:0x0F02 IND:0x0F02 [R] 16 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DEVICE_ID	15:0	0x5036	Two character ASCII code indicating device and configuration. For Rage 128 Pro the following device ID's are defined: 'PF' (5046h) = 385 BGA, AGP 4X, TMDS, PCI 'PR' (5052h) = 329 BGA, AGP 4x TMDS

DEVICE_ID: Device ID code

COMMAND CFG:0x0004 MMR:0x0F04 [R] MMR_1:0x0F04 [R] IND:0x0F04 [R] [RW] 16 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
IO_ACCESS_EN	0	0x0	<No Description> 0=Disable 1=Enable
MEM_ACCESS_EN	1	0x0	<No Description> 0=Disable 1=Enable
BUS_MASTER_EN	2	0x0	<No Description> 0=Disable 1=Enable

SPECIAL_CYCLE_EN (R)	3	0x0	<No Description> 0=Disable 1=Enable
MEM_WRITE_INVALIDATE_EN (R)	4	0x0	<No Description> 0=Disable 1=Enable
PAL_SNOOP_EN	5	0x0	<No Description> 0=Disable 1=Enable
PARITY_ERROR_EN (R)	6	0x0	<No Description> 0=Disable 1=Enable
AD_STEPPING (R)	7	0x1	<No Description>
SERR_EN (R)	8	0x0	<No Description>
FAST_B2B_EN	9	0x0	<No Description> 0=Disable 1=Enable
(reserved)	15:10		

COMMAND: <No Description>

STATUS CFG:0x0006 MMR:0x0F06 [R] MMR_1:0x0F06 [R] IND:0x0F06 [R] [RW] 16 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
CAP_LIST (R)	4	0x1	<No Description>
PCI_66_EN (R)	5	0x1	<No Description>

UDF_EN (R)	6	0x0	<No Description> 0=Disable 1=Enable
FAST_BACK_CAPABLE (R)	7	0x1	<No Description>
(reserved)	8		
DEVSEL_TIMING (R)	10:9	0x1	<No Description>
SIGNAL_TARGET_ABORT (R)	11	0x0	<No Description>
RECEIVED_TARGET_ABORT	12	0x0	<No Description> 0=Inactive 1=Active
RECEIVED_MASTER_ABORT	13	0x0	<No Description> 0=Inactive 1=Active
SIGNALED_SYSTEM_ERROR (R)	14	0x0	<No Description>
PARITY_ERROR_DETECTED (R)	15	0x0	<No Description>

STATUS: <No Description>

REVISION_ID CFG:0x0008 MMR:0x0F08 MMR_1:0x0F08 IND:0x0F08 **[R] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
MINOR_REV_ID	3:0	0x0	Incremented for minor revisions. Normally involving debugging, but not new significant features. 0000 = Initial Rage 128 revision.
MAJOR_REV_ID	7:4	0x0	Indicates major revisions within the Rage 128 family. Normally incremented when major features added. DEVICE_ID's are normally changed when MAJOR_REV_ID changes. 0000 = Intial version of Rage 128.

REVISION_ID: Indicates relative position of the device within the 'Rage 128' family.

REGPROG_INF CFG:0x0009 MMR:0x0F09 MMR_1:0x0F09 IND:0x0F09 [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
REG_LEVEL_PROG_INF	7:0	0x0	<No Description>

REGPROG_INF: <No Description>

SUB_CLASS CFG:0x000A MMR:0x0F0A MMR_1:0x0F0A IND:0x0F0A [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	6:0		
SUB_CLASS_INF	7	0x1	<No Description>

SUB_CLASS: <No Description>

BASE_CODE CFG:0x000B MMR:0x0F0B MMR_1:0x0F0B IND:0x0F0B [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BASE_CLASS_CODE	7:0	0x3	<No Description>

BASE_CODE: <No Description>

CACHE_LINE CFG:0x000C MMR:0x0F0C [R] MMR_1:0x0F0C [R] IND:0x0F0C [R] [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CACHE_LINE_SIZE	7:0	0x0	<No Description>

CACHE_LINE: <No Description>

LATENCY CFG:0x000D MMR:0x0F0D [R] MMR_1:0x0F0D [R] IND:0x0F0D [R] [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
LATENCY_TIMER	7:0	0x0	<No Description>

LATENCY: <No Description>

HEADER CFG:0x000E MMR:0x0F0E MMR_1:0x0F0E IND:0x0F0E [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
HEADER_TYPE	6:0	0x0	<No Description>
DEVICE_TYPE	7	0x0	<No Description> 0=Single-Function Device 1=Multi-Function Device

HEADER: <No Description>

BIST CFG:0x000F MMR:0x0F0F MMR_1:0x0F0F IND:0x0F0F [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BIST_COMP	3:0	0x0	<No Description>
(reserved)	5:4		
BIST_STRT	6	0x0	<No Description>
BIST_CAP	7	0x0	<No Description>

BIST: <No Description>

MEM_BASE CFG:0x0010 MMR:0x0F10 [R] MMR_1:0x0F10 [R] IND:0x0F10 [R] [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	2:0		
PREFETCH_EN (R)	3	0x1	<No Description>
(reserved)	25:4		
MEM_BASE	31:26	0x0	<No Description>

MEM_BASE: <No Description>

IO_BASE CFG:0x0014 MMR:0x0F14 [R] MMR_1:0x0F14 [R] IND:0x0F14 [R] [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BLOCK_IO_BIT (R)	7:0	0x1	<No Description> NOTE: Bits 7:1 of this field are hardwired to ZERO
IO_BASE	31:8	0x0	<No Description>

IO_BASE: <No Description>

REG_BASE CFG:0x0018 MMR:0x0F18 [R] MMR_1:0x0F18 [R] IND:0x0F18 [R] [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	13:0		
REG_BASE	31:14	0x0	<No Description>

REG_BASE: <No Description>

ADAPTER_ID CFG:0x002C MMR:0x0F2C MMR_1:0x0F2C IND:0x0F2C [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SUBSYSTEM_VENDOR_ID	15:0	STRAP	
SUBSYSTEM_ID	31:16	STRAP	

ADAPTER_ID: <No Description>

BIOS_ROM CFG:0x0030 MMR:0x0F30 [R] MMR_1:0x0F30 [R] IND:0x0F30 [R] [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BIOS_ROM_EN	0	0x0	Enable BIOS_ROM Define the bios_base_addr 0=Disable 1=Enable
(reserved)	16:1		
BIOS_BASE_ADDR	31:17	0x0	<No Description>

BIOS_ROM: <No Description>

CAPABILITIES_PTR CFG:0x0034 MMR:0x0F34 MMR_1:0x0F34 IND:0x0F34 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CAP_PTR	7:0	0x0	<No Description>
(reserved)	31:8		

CAPABILITIES_PTR: <No Description>

INTERRUPT_LINE CFG:0x003C MMR:0x0F3C [R] MMR_1:0x0F3C [R] IND:0x0F3C [R] [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
INTERRUPT_LINE	7:0	0xff	<No Description>

INTERRUPT_LINE: <No Description>

INTERRUPT_PIN CFG:0x003D MMR:0x0F3D MMR_1:0x0F3D IND:0x0F3D [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
INTERRUPT_PIN (R)	0	0x0	Indicates to system if device wants an interrupt resource. 0 = no interrupt wanted (strapped to disable interrupt). 1 = INTA# requested (strapped to enable interrupt).

(reserved)	7:1		

INTERRUPT_PIN: Interrupt resource request.

MIN_GRANT CFG:0x003E MMR:0x0F3E MMR_1:0x0F3E IND:0x0F3E [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MIN_GNT	7:0	0x8	<No Description>

MIN_GRANT: <No Description>

MAX_LATENCY CFG:0x003F MMR:0x0F3F MMR_1:0x0F3F IND:0x0F3F [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MAX_LAT	7:0	0x0	<No Description>

MAX_LATENCY: <No Description>

ADAPTER_ID_W CFG:0x004C [W] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SUBSYSTEM_VENDOR_ID	15:0	STRAP	<No Description>
SUBSYSTEM_ID	31:16	STRAP	<No Description>

ADAPTER_ID_W: <No Description>

CAPABILITIES_ID CFG:0x0050 MMR:0x0F50 MMR_1:0x0F50 IND:0x0F50 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CAP_ID	7:0	0x2	<No Description>
NEXT_PTR	15:8	0x5c	<No Description>
AGP_MINOR	19:16	0x0	<No Description>

AGP_MAJOR	23:20	0x2	<No Description>
(reserved)	31:24		

CAPABILITIES_ID: <No Description>

CONFIG_CNTL MMR:0x00E0 MMR_1:0x00E0 IOR:0x00E0 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
APER_0_ENDIAN	1:0	0x0	<No Description> 0=Little endian: (no swapping) 1=Big endian: 16 bpp swapping 2=Big endian: 32 bpp swapping
APER_1_ENDIAN	3:2	0x0	<No Description> 0=Little endian: (no swapping) 1=Big endian: 16 bpp swapping 2=Big endian: 32 bpp swapping
APER_REG_ENDIAN	4	0x0	<No Description> 0=Little endian: (no swapping) 1=Big endian: 32 bpp swapping
(reserved)	7:5		
CFG_VGA_RAM_EN	8	0x0	<No Description> 0=Disable 1=Enable
CFG_VGA_IO_DIS	9	STRAP	<No Description> 0=VGA I/O decode enabled if VGA_DISABLE@CONFIG_XSTRAP=0 1=VGA I/O decode disabled
(reserved)	15:10		
CFG_ATI_REV_ID (R)	19:16	0x0	<No Description>
(reserved)	31:20		

CONFIG_CNTL: Configuration Control

CONFIG_XSTRAP MMR:0x00E4 MMR_1:0x00E4 IOR:0x00E4 IND:0x00E4 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VGA_DISABLE (R)	0	0x0	<No Description>
BUS_CLK_SEL (R)	1	0x0	<No Description>
IDSEL (R)	2	0x0	<No Description>
ENINTB (R)	3	0x0	<No Description>
BUSTYPE (R)	5:4	0x0	<No Description>
AGPSKEW	7:6	STRAP	<No Description>
X1CLK_SKEW	9:8	STRAP	<No Description>
FLASH_ROM (R)	10	0x0	<No Description>
SEPROM_EN (R)	11	0x0	<No Description>
VIP_DEVICE (R)	12	0x0	<No Description>
ADDIN_CARD (R)	13	0x0	<No Description>
(reserved)	31:14		

CONFIG_XSTRAP: <No Description>

CONFIG_BONDS MMR:0x00E8 MMR_1:0x00E8 IOR:0x00E8 IND:0x00E8 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
RSTRAP (R)	1:0	0x0	<No Description>
PKGTYPE_0 (R)	2	0x0	<No Description>
CRIPPLEb (R)	3	0x0	<No Description>
STRSTb (R)	4	0x0	<No Description>
(reserved)	5		

AVCOGN (R)	6	0x0	<No Description>
PKGTYPE_1 (R)	7	0x0	<No Description>
CRIPPLEAGP4X (R)	8	0x0	<No Description>
CRIPPLEPANEL (R)	9	0x0	<No Description>
(reserved)	31:10		

CONFIG_BONDS: <No Description>

CONFIG_MEMSIZE MMR:0x00F8 MMR_1:0x00F8 IOR:0x00F8 IND:0x00F8 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CONFIG_MEMSIZE	25:0	0x0	Size of the frame buffer in bytes. Includes embedded memory if present. NOTE: Bits 20:0 of this field are hardwired to ZERO
(reserved)	31:26		

CONFIG_MEMSIZE: Frame Buffer Size

CONFIG_APER_0_BASE MMR:0x0100 MMR_1:0x0100 IND:0x0100 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	25:0		
APER_0_BASE	31:26	0x0	Base address of image 0 of the linear aperture.

CONFIG_APER_0_BASE: Linear Aperture 0 Base

CONFIG_APER_1_BASE MMR:0x0104 MMR_1:0x0104 IND:0x0104 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	24:0		

APER_1_BASE	31:25	0x0	Base address of image 1 of the linear aperture. Both the first and second linear apertures function the same. The second aperture is mainly for use on PowerMac systems, where each aperture can have its bi-endian swapping set independently (see CONFIG_CNTL). NOTE: Bit 0 of this field is hardwired to ONE

CONFIG_APER_1_BASE: Linear Aperture 1 Base

3.3 General VGA Status and Configuration

<No description>

GENMO_WT VGA_IO:0x03C2 [W] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B	0	0x0	Emulation addressing mode. 0=Monochrome emulation, regs at 0x3Bx 1=Color/Graphic emulation, regs at 0x3Dx
VGA_RAM_EN	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture. 0=Disable 1=Enable
VGA_CKSEL	3:2	0x0	Selects pixel clock frequency to use. 0=25.1744MHz (640 Pels) 1=28.3212MHz (720 Pels) 2=Reserved 3=Reserved
(reserved)	4		
ODD_EVEN_MD_PGSEL	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	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low VGA convention uses active-low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.

VGA_VSYNC_POL	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low VGA convention uses active-high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

GENMO_WT:
Output Register (Write)

GENMO_RD VGA_IO:0x03CC [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GENMO_MONO_ADDRESS_B	0	0x0	Emulation addressing mode. 0=Monochrome emulation, regs at 0x3Bx 1=Color/Graphic emulation, regs at 0x3Dx
VGA_RAM_EN	1	0x0	Enables/Disables CPU access to video RAM at VGA aperture. 0=Disable 1=Enable
VGA_CKSEL	3:2	0x0	Selects pixel clock frequency to use. 0=25.1744MHz (640 Pels) 1=28.3212MHz (720 Pels) 2=Reserved 3=Reserved
(reserved)	4		
ODD_EVEN_MD_PGSEL	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	6	0x0	Determines polarity of horizontal sync (HSYNC) for VGA modes. 0 = HSYNC pulse active high 1 = HSYNC pulse active low VGA covnention uses active-low VSYNC for 400 (and 200) and 480 line modes. Active high is normally used for 350 line modes.
VGA_VSYNC_POL	7	0x0	Determines polarity of vertical sync (VSYNC) for VGA modes. 0 = VSYNC pulse active high 1 = VSYNC pulse active low VGA convention uses active-high VSYNC for 400 (and 200) line modes. Active low is normally used for 350 and 480 line modes.

GENMO_RD: Miscellaneous Output Register (Read)

GENFC_RD VGA_IO:0x03CA [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	2:0		
VSYNC_SEL	3	0x0	Vertical sync select (read). 0=Normal vertical sync 1=Sync is 'vertical sync' ORed with 'vertical display enable'
(reserved)	7:4		

GENFC_RD: Feature Control Register (Read)

GENFC_WT VGA_IO:0x03BA VGA_IO:0x03DA [W] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	2:0		
VSYNC_SEL	3	0x0	Vertical sync select (write). 0=Normal vertical sync 1=Sync is 'vertical sync' ORed with 'vertical display enable'
(reserved)	7:4		

GENFC_WT: Feature Control Register (Read)

GENS0 VGA_IO:0x03C2 [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
SENSE_SWITCH	4	0x0	DAC comparator read back. Used for monitor detection. Mirror of DAC_CMP_OUTPUT@DAC_CNTL. See description there.
(reserved)	6:5		
CRT_INTR	7	0x0	CRT Interrupt: 0=Vertical retrace interrupt is cleared 1=Vertical retrace interrupt is pending

GENS0: Input Status 0 Register

GENS1 VGA_IO:0x03BA VGA_IO:0x03DA [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
NO_DIPLAY	0	0x0	 0=Enable 1=Disable
(reserved)	2:1		
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 color 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

(reserved)	7:6		

GENS1: Input Status 1 Register

GENENB VGA_IO:0x03C3 [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BLK_IO_BASE	7:0	0x0	Readback of block I/O aperture base offset. Mirror of the PCI configuration space register. Used here so software can find the apertures if they are relocated by the OS.

GENENB: Block I/O Base

3.4 VGA DAC Registers

<No description>

DAC_DATA VGA_IO:0x03C9 [RW] 8 bits (access: 8/16/32)			
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.

DAC_DATA: VGA Palette (DAC) Data

DAC_MASK VGA_IO:0x03C6 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DAC_MASK	7:0	0xff	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.

DAC_MASK: Palette index mask for VGA emulation modes.

DAC_R_INDEX VGA_IO:0x03C7 [RW] 8 bits (access: 8/16/32)			
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). Also refer to DAC_W_INDEX.

DAC_R_INDEX: Palette (DAC) Read Index

DAC_W_INDEX VGA_IO:0x03C8 [RW] 8 bits (access: 8/16/32)			
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. Also refer to DAC_R_INDEX.

DAC_W_INDEX: Palette (DAC) Write Index

3.5 VGA Sequencer Registers

<No description>

SEQ8_IDX VGA_IO:0x03C4 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SEQ_IDX	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.
(reserved)	7:3		

SEQ8_IDX: <No Description>

SEQ8_DATA VGA_IO:0x03C5 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SEQ_DATA	7:0	0x0	<No Description>

SEQ8_DATA: <No Description>

SEQ00 SEQ:0x0000 [RW] 8 bits (access: 8/16/32)			
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
(reserved)	7:2		

SEQ00: Reset Register

SEQ01 SEQ:0x0001 [RW] 8 bits (access: 8/16/32)			
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.
(reserved)	1		
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
(reserved)	7:6		

SEQ01: Clock Mode Register

SEQ02 SEQ:0x0002 [RW] 8 bits (access: 8/16/32)			
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
(reserved)	7:4		

SEQ02: Map Mask Register

SEQ03 SEQ:0x0003 [RW] 8 bits (access: 8/16/32)			
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

(reserved)	7:6		

SEQ03: Character Map Select Register

SEQ04 SEQ:0x0004 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	0		
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 the only 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
(reserved)	7:4		

SEQ04: Memory Mode Register

3.6 VGA CRT Registers

<No description>

CRTC8_IDX VGA_IO:0x03B4 VGA_IO:0x03D4 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VCRTC_IDX	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.
(reserved)	7:6		

CRTC8_IDX: CRT Index Register

CRTC8_DATA VGA_IO:0x03B5 VGA_IO:0x03D5 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VCRTC_DATA	7:0	0x0	CRTC data indirect access

CRTC8_DATA: CTRC Data Register

CRT00 CRT:0x0000 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
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.

CRT00: Horizontal Total Register

CRT01 CRT:0x0001 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_DISP_END	7:0	0x0	These bits define the active horizontal display in a scan line. The value is one less than the total number of displayed characters in a scan line.

CRT01: Horizontal Display Enable End Register

CRT02 CRT:0x0002 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_START	7:0	0x0	These bits define the horizontal character count that represents the character count in the active display area plus the right border. In other words, the count is from the start of active display to the start of triggering of the H blanking pulse.

CRT02: Start Horizontal Blanking Register

CRT03 CRT:0x0003 [RW] 8 bits (access: 8/16/32)			
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: (in characters) 0=0Skew 1=1Skew 2=2Skew 3=3Skew
CR10CR11_R_DIS_B	7	0x0	Compatibility Read: 0=WrtOnlyToCRT10-11 1=WrtRdToCRT10-11

CRT03: End Horizontal Blanking Register

CRT04 CRT:0x0004 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_START	7:0	0x0	These bits define the horizontal character count at which the horizontal retrace pulse becomes active.

CRT04: Start Horizontal Retrace Register

CRT05 CRT:0x0005 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_END	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 blanking 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 in CRT03[4:0].

CRT05: End Horizontal Retrace Register

CRT06 CRT:0x0006 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
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.

CRT06: Vertical Total Register

CRT07 CRT:0x0007 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL_B8	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	1	0x0	End V Display Bit 8 (CRT12). Bit 8 of 10-bit vertical count for V Display enable. For functional description, refer to CRT12 register.

V_SYNC_START_B8	2	0x0	Start V Retrace Bit 8 (CRT10). Bit 8 of 10-bit vertical 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	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	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	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	7	0x0	Start V Retrace Bit (CRT10). Bit 9 of 10-bit vertical count for V Retrace start. For functional description see CRT10 register.

CRT07: CRTC Overflow Register

CRT08 CRT:0x0008 [RW] 8 bits (access: 8/16/32)			
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).
(reserved)	7		

CRT08: Preset Row Scan Register

CRT09 CRT:0x0009 [RW] 8 bits (access: 8/16/32)			
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 vertical count for line compare. For functional description see CRT18 register.
LINE_CMP_B9	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

CRT09: Maximum Scan Line Register

CRT0A CRT:0x000A [RW] 8 bits (access: 8/16/32)			
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
(reserved)	7:6		

CRT0A: Cursor Start Register

CRT0B CRT:0x000B [RW] 8 bits (access: 8/16/32)			
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.
(reserved)	7		

CRT0B: Cursor End Register

CRT0C CRT:0x000C [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DISP_START	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.

CRT0C: Start Address (High Byte) Register

CRT0D CRT:0x000D [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

DISP_START	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. For split-screen mode, CRT0C + CRT0D points to the starting location of screen A (top half.) The starting address for screen B is always zero.

CRT0D: Start Address (Low Byte) Register

CRT0E CRT:0x000E **[RW] 8 bits (access: 8/16/32)**			
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.

CRT0E: Cursor Location (High Byte) Register

CRT0F CRT:0x000F **[RW] 8 bits (access: 8/16/32)**			
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

CRT0F: Cursor Location (Low Byte) Register

CRT10 CRT:0x0010 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**

V_SYNC_START	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.

CRT10: Start Vertical Retrace Register

CRT11 CRT:0x0011 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_END	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	<No Description> 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

CRT11: End Vertical Retrace Register

CRT12 CRT:0x0012 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

V_DISP_END	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.

CRT12: Vertical Display Enable End Register

CRT13 CRT:0x0013 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DISP_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.

CRT13: Offset Register

CRT14 CRT:0x0014 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
UNDRLN_LOC	4:0	0x0	H Row Scan Bits 4-0.- These bits define the horizontal scan row, from the top of the character line, that should be used for underlining. The 5-bit value is equal to the actual number minus one.
ADDR_CNT_BY4	5	0x0	Count-by-4: 0=Char. Clock 1=CountBy4
DOUBLE_WORD	6	0x0	Double-Word Mode: 0=Disable 1=DoubleWordMdEna
(reserved)	7		

CRT14: Underline Location Register

CRT15 CRT:0x0015 [RW] 8 bits (access: 8/16/32)			
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 blanking pulse, in units of horizontal scan lines. The value is equal to the actual number of displayed lines minus one.

CRT15: Start Vertical Blanking Register

CRT16 CRT:0x0016 [RW] 8 bits (access: 8/16/32)			
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 stored 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.

CRT16: End Vertical Blanking Register

CRT17 CRT:0x0017 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
RA0_AS_A13B	0	0x0	Compatibility Mode:
RA1_AS_A14B	1	0x0	Select Row Scan Counter:
VCOUNT_BY2	2	0x0	Vertical_by_2 NOTE: When bit 2 is logical one, other vertical register values should be adjusted as well (CRT06, CRT10, CRT12, CRT15, and CRT18).
ADDR_CNT_BY2	3	0x0	Count_by_2: ENGINEERING NOTE: Bit can be written and read, but has no effect.
(reserved)	4		

WRAP_A15TOA0	5	0x0	Address Wrap: ENGINEERING NOTE: Bit can be written and read, but has no effect.
BYTE_MODE	6	0x0	Byte/Word Mode: 0=WordMode 1=ByteMode
CRTC_SYNC_EN	7	0x0	H/V Retrace Enable: 0=Disable HVSync 1=EnaHVSync

CRT17: CRT Mode Register

CRT18 CRT:0x0018 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
LINE_CMP	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].)

CRT18: Line Compare Register

CRT1E CRT:0x001E **[R] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
(reserved)	0		
GRPH_DEC_RD1	1	0x0	This register is used to read back the graphics controller index decode.
(reserved)	7:2		

CRT1E: Graphics Controller Index Decode Register

CRT1F CRT:0x001F [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_DEC_RD0	7:0	0x0	This register is used to read back the graphics controller index decode.

CRT1F: Graphics Controller Index Decode Register

CRT22 CRT:0x0022 [R] 8 bits (access: 8/16/32)			
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.

CRT22: RAM Data Latch Readback Register

CRT00_S CRT:0x0040 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_TOTAL_S	7:0	0x0	Flat Panel Shadow version of Horizontal Total, the number of active characters on a scan line, including the retrace period. The value is five less than the total number of displayed characters in a scan line.

CRT00_S: <No Description>

CRT01_S CRT:0x0041 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_DISP_END_S	7:0	0x0	Flat Panel Shadow version of Horizontal Display End. Defines the number of characters in the active display on one scan line. The value is one less than the total number of displayed characters in a scan line.

CRT01_S: <No Description>

CRT02_S CRT:0x0042 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_START_S	7:0	0x0	Flat Panel Shadow version of Horizontal Blank Start. Defines the horizontal character count that represents the character count in the active display area plus the right border. In other words, the count is from the start of the active display to the start of triggering of the H blanking pulse.

CRT02_S: <No Description>

CRT03_S CRT:0x0043 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_BLANK_END_S	4:0	0x0	Flat Panel Shadow version of Horizontal Blank End bits 4 to 0. These are the five low-order bits (six bits total) of horizontal character count for triggering the end of the horizontal blanking pulse.
H_DE_SKEW_S	6:5	0x0	Flat Panel Shadow version of the Horizontal Display Enable Skew (in characters) 0=0Skew 1=1Skew 2=2Skew 3=3Skew
CR10CR11_R_DIS_B_M	7	0x0	<No Description> 0=WrtOnlyToCRT10-11 1=WrtRdToCRT10-11

CRT03_S: <No Description>

CRT04_S CRT:0x0044 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_START_S	7:0	0x0	Flat Panel Shadow version of the Horizontal Sync Start. Defines the horizontal character count at which the horizontal retrace pulse becomes active.

CRT04_S: <No Description>

CRT05_S CRT:0x0045 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
H_SYNC_END_S	4:0	0x0	Flat Panel Shadow version of Horizontal Sync End. The value in this register contains the 5-bit result from the sum of CRT0 plus the width of the horizontal retrace pulse, in character clock units
H_SYNC_SKEW_S	6:5	0x0	Flat Panel Shadow version of Horizontal Retrace Delay. These two bits delay the horizontal retrace pulse the specified number of character clocks
H_BLANK_END_B5_S	7	0x0	Flat Panel Shadow version of bit 5 of Horizontal Blank. This is the MSB of the 6-bit character count for the H Blanking End Pulse. The other five low-order bits are in CRT03_S[4:0]

CRT05_S: <No Description>

CRT06_S CRT:0x0046 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL_S	7:0	0x0	Flat Panel Shadow version of the Vertical Total. These are the eight low-order bits of 10-bit vertical total register. The 2 high-order bits are CRT07_S[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.

CRT06_S: <No Description>

CRT07_S CRT:0x0047 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_TOTAL_B8_S	0	0x0	Flat Panel Shadow version of Vertical Total Bit 8 (CRT06_S) (10 bits in length). For functional description see CRT06_S register
V_DISP_END_B8_S	1	0x0	Flat Panel Shadow version of Vertical Display End Bit 8 (CRT12_S) (10 bits in length). For functional description see CRT12_S register
V_SYNC_START_B8_S	2	0x0	Flat Panel Shadow version of Vertical Sync Start Bit 8 (CRT10_S) (10 bits in length). For functional description see CRT10_S register

V_BLANK_START_B8_S	3	0x0	Flat Panel Shadow version of Vertical Blank Start Bit 8 (CRT15_S) (10 bits in length). For functional description see CRT15_S register
LINE_CMP_B8_M	4	0x0	Flat Panel Shadow version of Vertical Count for Line Compare bit 8 (CRT18) (10 bits in length). For functional description see CRT18_S register
V_TOTAL_B9_S	5	0x0	Flat Panel Shadow version of Vertical Total Bit 9 (CRT06_S) (10 bits in length). See CRT06_S for functional description
V_DISP_END_B9_S	6	0x0	Flat Panel Shadow version of Vertical Display End Bit 9 (CRT12_S) (10 bits in length). See CRT12_S for functional description
V_SYNC_START_B9_S	7	0x0	Flat Panel Shadow version of Vertical Sync Start Bit 9 (CRT15_S) (10 bits in length). See CRT15_S for functional description

CRT07_S: <No Description>

CRT08_S CRT:0x0048 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
ROW_SCAN_START_M	4:0	0x0	Flat Panel Shadow version of Preset Row Scan bits 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 Vertical Retrace (in the next frame). Each H Retrace pulse increments the counter by 1, up to the maximum scan line value programmed by CRT09_S, then the counter is cleared
BYTE_PAN_M	6:5	0x0	Flat Panel Shadow version of 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 see ATTR13).
(reserved)	7		

CRT08_S: <No Description>

CRT09_S CRT:0x0049 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**

MAX_ROW_SCAN_S	4:0	0x0	Flat Panel Shadow version of Maximum Scan Line Bits. These bits define a value that is the actual number of scan lines per character minus 1
V_BLANK_START_B9_S	5	0x0	Flat Panel Shadow version of Start Vertical Blanking bit 9 (CRT15_S) (10 bits in length). For functional description see CRT18_S register
LINE_CMP_B9_M	6	0x0	<No Description>
DOUBLE_CHAR_HEIGHT_M	7	0x0	<No Description> 0=200LineScan 1=400LineScan

CRT09_S: <No Description>

CRT0A_S CRT:0x004A **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
CURSOR_START_S	4:0	0x0	<No Description>
CURSOR_DISABLE_M	5	0x0	<No Description> 0=on 1=off
(reserved)	7:6		

CRT0A_S: <No Description>

CRT0B_S CRT:0x004B **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
CURSOR_END_S	4:0	0x0	<No Description>
CURSOR_SKEW_M	6:5	0x0	<No Description>
(reserved)	7		

CRT0B_S: <No Description>

CRT0C_S CRT:0x004C [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DISP_START_M	7:0	0x0	<No Description>

CRT0C_S: <No Description>

CRT0D_S CRT:0x004D [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DISP_START_M	7:0	0x0	<No Description>

CRT0D_S: <No Description>

CRT0E_S CRT:0x004E [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CURSOR_LOC_HI_M	7:0	0x0	<No Description>

CRT0E_S: <No Description>

CRT0F_S CRT:0x004F [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CURSOR_LOC_LO_M	7:0	0x0	<No Description>

CRT0F_S: <No Description>

CRT10_S CRT:0x0050 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_START_S	7:0	0x0	<No Description>

CRT10_S: <No Description>

CRT11_S CRT:0x0051 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_SYNC_END_S	3:0	0x0	<No Description>
V_INTR_CLR_M	4	0x0	<No Description> 0=VRetraceIntCleared 1=Not Cleared
V_INTR_EN_M	5	0x0	<No Description> 0=VRetraceIntEna 1=Disable
SEL5_REFRESH_CYC_M	6	0x0	<No Description> 0=3 DRAM Refresh/Horz Line 1=5 DRAM Refresh/Horz Line
C0T7_WR_ONLY_M	7	0x0	<No Description> 0=EnaWrtToCRT00-07 1=C0T7B4WrtOnly

CRT11_S: <No Description>

CRT12_S CRT:0x0052 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_DISP_END_S	7:0	0x0	<No Description>

CRT12_S: <No Description>

CRT13_S CRT:0x0053 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

DISP_PITCH_M	7:0	0x0	<No Description>

CRT13_S: <No Description>

CRT14_S CRT:0x0054 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
UNDRLN_LOC_S	4:0	0x0	<No Description>
ADDR_CNT_BY4_M	5	0x0	<No Description> 0=Char. Clock 1=CountBy4
DOUBLE_WORD_M	6	0x0	<No Description> 0=Disable 1=DoubleWordMdEna
(reserved)	7		

CRT14_S: <No Description>

CRT15_S CRT:0x0055 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_BLANK_START_S	7:0	0x0	<No Description>

CRT15_S: <No Description>

CRT16_S CRT:0x0056 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
V_BLANK_END_S	7:0	0x0	<No Description>

CRT16_S: <No Description>

CRT17_S CRT:0x0057 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

RA0_AS_A13B_M	0	0x0	<No Description>
RA1_AS_A14B_M	1	0x0	<No Description>
VCOUNT_BY2_S	2	0x0	<No Description>
ADDR_CNT_BY2_M	3	0x0	<No Description>
(reserved)	4		
WRAP_A15TOA0_M	5	0x0	<No Description>
BYTE_MODE_M	6	0x0	<No Description> 0=WordMode 1=ByteMode
CRTC_SYNC_EN_M	7	0x0	<No Description> 0=Disable HVSync 1=EnaHVSync

CRT17_S: <No Description>

CRT18_S CRT:0x0058 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
LINE_CMP_M	7:0	0x0	<No Description>

CRT18_S: <No Description>

CRT1E_S CRT:0x005E **[R] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
(reserved)	0		
GRPH_DEC_RD1_M	1	0x0	<No Description>
(reserved)	7:2		

CRT1E_S: <No Description>

CRT1F_S CRT:0x005F [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_DEC_RD0_M	7:0	0x0	<No Description>

CRT1F_S: <No Description>

CRT22_S CRT:0x0062 [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_LATCH_DATA_M	7:0	0x0	<No Description>

CRT22_S: <No Description>

CRTC_DEBUG MMR:0x021C MMR_1:0x021C IND:0x021C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_GUI_TRIG_BYPASS_EN	0	0x0	0=Don't bypass gui triggers generated by dispeng 1=bypass gui triggers generated by dispeng
GUI_TRIG_VLINE_BYPASS	1	0x0	<No Description>
GUI_TRIG_OFFSET_BYPASS	2	0x0	<No Description>
GUI_TRIG_PITCH_ADD_BYPASS	3	0x0	<No Description>
(reserved)	31:4		

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

3.7 VGA Graphics Registers

<No description>

GRPH8_IDX VGA_IO:0x03CE [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_IDX	3:0	0x0	This index is used to address one of the internal registers of the graphics controller (GRAC) at I/O port 0x3CRF.
(reserved)	7:4		

GRPH8_IDX: <No Description>

GRPH8_DATA VGA_IO:0x03CF [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_DATA	7:0	0x0	<No Description>

GRPH8_DATA: <No Description>

GRA00 GRPH:0x0000 [RW] 8 bits (access: 8/16/32)			
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:
(reserved)	7:4		

GRA00: Set/Reset Register

GRA01 GRPH:0x0001 [RW] 8 bits (access: 8/16/32)			
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:
(reserved)	7:4		

GRA01: Enable Set/Reset Register

GRA02 GRPH:0x0002 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_CCOMP	3:0	0x0	Color 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 color 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 Color 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.
(reserved)	7:4		

GRA02: Color Compare Register

GRA03 GRPH:0x0003 [RW] 8 bits (access: 8/16/32)			
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 subsequent 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 functions 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
(reserved)	7:5		

GRA03: Data Rotate Register

GRA04 GRPH:0x0004 [RW] 8 bits (access: 8/16/32)			
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 (useful in transferring bit map data between the maps and system RAM).
(reserved)	7:2		

GRA04: Read Map Select Register

GRA05 GRPH:0x0005 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_WRITE_MODE	1:0	0x0	Write Mode: 0=Write mode 0 1=Write mode 1 2=Write mode 2 3=Write mode 3
(reserved)	2		
GRPH_READ1	3	0x0	Read Mode: 0=Read mode 0, byte oriented 1=Read mode 1, pixel oriented

CGA_ODDEVEN	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	5	0x0	Shift Register Mode: This bit controls how data form 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	6	0x0	256 Color Mode. This bit also controls 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
(reserved)	7		

GRA05: Graphics Mode Register

GRA06 GRPH:0x0006 **[RW] 8 bits (access: 8/16/32)**			
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
(reserved)	7:4		

GRA06: Graphics Miscellaneous Register

GRA07 GRPH:0x0007 [RW] 8 bits (access: 8/16/32)			
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
(reserved)	7:4		

GRA07: Color Don't Care Register

GRA08 GRPH:0x0008 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
GRPH_BMSK	7:0	0x0	Bit Mask:

GRA08: Bit Mask Register

3.8 VGA Attribute Registers

<No description>

ATTRX VGA_IO:0x03C0 [RW] 8 bits (access: 8/16/32)			
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 can 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 color palette, this bit should be set to logical 1. 0=Processor to load 1=Memory data to access
(reserved)	7:6		

ATTRX: Attribute Index Register

ATTRDW VGA_IO:0x03C0 [W] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
ATTR_DATA	7:0	0x0	Attribute Data Write

ATTRDW: Attribute Data Write Register

ATTRDR VGA_IO:0x03C1 [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
ATTR_DATA	7:0	0x0	Attribute Data Read

ATTRDR: Attribute Data Read Register

ATTR00 ATTR:0x0000 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
ATTR_PAL0	5:0	0x0	Color Bits 5:0 map the text attribute or graphics color input value to a display color on the screen. Color is disabled for those bits that are set to logical 0; enabled for those bits set to logical 1.
(reserved)	7:6		

ATTR00: Palette Register 0

ATTR00 to ATTR0F are all identical.

ATTR10 ATTR:0x0010 [RW] 8 bits (access: 8/16/32)			
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/Color 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 color. 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
(reserved)	4		
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 color), 8 bits packed to form a pixel
ATTR_CSEL_EN	7	0x0	Alternate Color Source: 0=Select ATTR00-0F bit 5:4 as P5 and P4 1=Select ATTR14 bit 1:0 as P5 and P4

ATTR10: Mode Control Register

ATTR11 ATTR:0x0011 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
ATTR_OVSC	7:0	0x0	Overscan Color

ATTR11: Overscan Color Register

ATTR12 ATTR:0x0012 **[RW] 8 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
ATTR_MAP_EN	3:0	0x0	Enable Color 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 color 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
(reserved)	7:6		

ATTR12: Color Map Enable Register

ATTR13 ATTR:0x0013 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
ATTR_PPAN	3:0	0x0	Shift Count Bits 3:0. The shift count value (0-8) indicates how many pixel 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 - -
(reserved)	7:4		

ATTR13: Horizontal PEL Panning Register

ATTR14 ATTR:0x0014 [RW] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
ATTR_CSEL1	1:0	0x0	Color bits P5 and P4, respectively. These are the color output bits (instead of bits 5 and 4 of the internal palette registers ATTR00-0F) when alternate color source, bit ATTR10[7] is logical 1.

ATTR_CSEL2	3:2	0x0	Color bits P7 and P6, respectively. These two bits are the two high-order bits of the 8-bit color, used for rapid color set switching (addressing different parts of the DAC color lookup table). The lower order bits are in registers ATTR00-0F.
(reserved)	7:4		

ATTR14: Color Select Register

3.9 CRTC

The CRTC generates the horizontal sync, vertical sync, and blank signals used to position the pixel data on the display monitor. All horizontal parameters are in terms of characters (pixels * 8). All vertical paramters are in terms of lines. Accurate display centering is possible by adjusting CRTC_HORZ_SYNC_DLY. A vertical blank and vertical line interrupt allows video synchronization without motion tearing artifacts. Monitor power management is controlled through CRTC_HSYNC_DIS and CRTC_VSYNC_DIS.

CRTC_GEN_CNTL MMR:0x0050 MMR_1:0x0050 IOR:0x0050 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_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
CRTC_INTERLACE_EN	1	0x0	Interlace enable. 0=Non-Interlace 1=Interlace
(reserved)	3:2		
CRTC_C_SYNC_EN	4	0x0	Enables composite sync on horizontal sync output. 0=Disable 1=Enable
(reserved)	7:5		

CRTC_PIX_WIDTH	10:8	0x0	Display pixel width (actually depth): 1=4bpp 2=8bpp 3=15bpp 4=16bpp 5=24bpp 6=32bpp
(reserved)	15:11		
CRTC_CUR_EN	16	0x0	Hardware Cursor Enable: 0=Disable 1=Enable
CRTC_CUR_MODE	19:17	0x0	Hardware Cursor Mode: 0 = 2bpp monochrome 64x64. 2 color, transparent, inverse. others = reserved for future use. 0=VGA_VSTATUS until vcount= vtotal, DISP_ADDR loads when vcount=vtotal 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 4=VGA_VSTATUS until vcount= vtotal,DISP_ADDR loads when vcount=vtotal 5=VGA_VSTATUS until vblank end, DISP_ADDR loads when vcount=vtotal 6=VGA_VSTATUS until vcount= vtotal, DISP_ADDR loads in vsync start 7=VGA_VSTATUS until vblank end, DISP_ADDR loads in vsync start
(reserved)	23:20		
CRTC_EXT_DISP_EN	24	0x0	Extended display mode enable: (default = 0) 0=VGA 1=Extended
CRTC_EN	25	0x0	Enables CRT controller: (default = 0) 0=Reset 1=Enable

CRTC_DISP_REQ_EN_B	26	0x1	Display request to memory controller enable: Active low 0=Enable 1=Disable
(reserved)	31:27		

CRTC_GEN_CNTL: CRTC General Controls

CRTC_EXT_CNTL MMR:0x0054 MMR_1:0x0054 IOR:0x0054 IND:0x0054 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_VGA_XOVERSCAN	0	0x0	VGA Overscan: 0=Disable extended overscan in VGA 1=Enable extended overscan in VGA
VGA_BLINK_RATE	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	3	0x0	Enable linear addressing through VGA aperture 0=Disable 1=Enable
VGA_128KAP_PAGING	4	0x0	Enable extended aperture paging in 128K VGA aperture mode: 0=Normal 1=Enable
VGA_TEXT_132	5	0x0	Extended text mode select (linear address 132 column text mode) 0=inActive 1=Active

VGA_XCRT_CNT_EN	6	0x0	Extended CRTC display address counter enable. Active High 0=Disable 1=Enable Ext CRTC Counter
(reserved)	7		
CRTC_HSYNC_DIS	8	0x0	Disables horizontal sync output. To be used for DPMS signaling. 0=Enable 1=Disable
CRTC_VSYNC_DIS	9	0x0	Disables vertical sync output. To be used for DPMS signaling. Note this must remain enabled while using composite SYNC on HSYNC (CRTC_C_SYNC_EN=1). 0=Enable 1=Disable
CRTC_DISPLAY_DIS	10	0x0	Disables the display, forcing the blanking signal to be active. 0=Enable 1=Blanked
CRTC_SYNC_TRISTATE	11	0x0	Sync Tristate Enable: 0=Normal 1=Tristate HSYNC and VSYNC outputs
CRTC_HSYNC_TRISTATE	12	0x0	<No Description> 0=Normal HSYNC 1=Tristate HSYNC output
CRTC_VSYNC_TRISTATE	13	0x0	<No Description> 0=Normal VSYNC 1=Tristate VSYNC output
(reserved)	16:14		
VGA_CUR_B_TEST	17	0x0	Test cursor blinking. Active High. 0=Disable VGA cursor test 1=Test VGA cursor blinking

VGA_PACK_DIS	18	0x0	Controls host write pipe for packed VGA modes (e.g. mode 13) 0=Fast VGA write in packed modes 1=Normal VGA write in packed modes
VGA_MEM_PS_EN	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
VGA_READ_PREFETCH_DIS	20	0x0	VGA read pre-fetching control: 0=Prefetch VGA read data for next byte after each read. 1=Disable VGA read prefetching.
DFIFO_EXTSENSE	21	0x1	Extended Sensing control for display FIFO macro. '1' is safer. BIOS will set.
FP_OUT_EN	22	0x0	Flat Panel output control. 0=Tri-state digital flat panel outputs 1=Enable flat panel digital outputs
FP_ACTIVE	23	0x0	Flat panel strap override. Setting low will return pins to other operation if strapped for panel. No affect if not strapped for panel operation. 0=Flat panel digital outputs not set to panel function 1=Flat panel digital outputs set to panel data from RAMDAC
VCRTC_IDX_MASTER	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.
(reserved)	31		

CRTC_EXT_CNTL: Extended General CRTC Controls

CRTC_STATUS MMR:0x005C MMR_1:0x005C IOR:0x005C IND:0x005C [RW] 32 bits (access: 8/16/32)			
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_CLEAR (W)	1	0x0	Used to clear CRTC_VBLANK_SAVE. 0=No effect 1=Clear CRTC_VBLANK_SAVE
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) occurred. 0=No vertical blank since last clear 1=Vertical Blank since last cleared
CRTC_VLINE_SYNC (R)	2	0x0	Indicates if the scan line is even (0) or odd (1) 0=Even scan line 1=Odd scan line
CRTC_FRAME (R)	3	0x0	Indicates if even (1) or odd (0) frame currently displayed. 0: even 0=Even frame 1=Odd frame
(reserved)	30:4		
FIX_VSYNC_TIMING	31	0x1	<No Description> 0=Standard VSYNC output timing 1=Modified VSYNC output timing(1 clk cycle later)

CRTC_STATUS: Status bits to determine current state of the display.

CRTC_H_TOTAL_DISP MMR:0x0200 MMR_1:0x0200 IND:0x0200 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

CRTC_H_TOTAL	8:0	0x0	Horizontal total (pixels * 8). Sum of display width, overscan right, front porch, sync width, back porch and overscan left.
(reserved)	15:9		
CRTC_H_DISP	23:16	0x0	Horizontal display end (pixels * 8). Determines number of visible pixels, not including overscan.
(reserved)	31:24		

CRTC_H_TOTAL_DISP: Horizontal Total Control

CRTC_H_SYNC_STRT_WID MMR:0x0204 MMR_1:0x0204 IND:0x0204 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_H_SYNC_STRT_PIX	2:0	0x0	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	11:3	0x0	Horizontal sync start (pixels * 8). Sum of display width, overscan right and front porch.
(reserved)	15:12		
CRTC_H_SYNC_WID	21:16	0x0	Horizontal sync width (pixels * 8)
(reserved)	22		
CRTC_H_SYNC_POL	23	0x0	Horizontal sync polarity 1 = Negative Polarity (active low) 0 = Positive Polarity (active high) 0=Positive 1=Negative
(reserved)	31:24		

CRTC_H_SYNC_STRT_WID: Horizontal Sync Control

CRTC_V_TOTAL_DISP MMR:0x0208 MMR_1:0x0208 IND:0x0208 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

CRTC_V_TOTAL	10:0	0x0	Vertical total. Sum of display height, overscan bottom, front porch, sync width, back porch and overscan top.
(reserved)	15:11		
CRTC_V_DISP	26:16	0x0	Vertical display end. Determines number of visible lines, not including overscan.
(reserved)	31:27		

CRTC_V_TOTAL_DISP: Vertical Total Control

CRTC_V_SYNC_STRT_WID MMR:0x020C MMR_1:0x020C IND:0x020C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_V_SYNC_STRT	10:0	0x0	Vertical sync start. Sum of display height, overscan bottom and front porch.
(reserved)	15:11		
CRTC_V_SYNC_WID	20:16	0x0	Vertical sync width
(reserved)	22:21		
CRTC_V_SYNC_POL	23	0x0	Vertical sync polarity 0=Positive 1=Negative
(reserved)	31:24		

CRTC_V_SYNC_STRT_WID: Vertical Sync Control

CRTC_VLINE_CRNT_VLINE MMR:0x0210 MMR_1:0x0210 IND:0x0210 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_VLINE	10:0	0x0	Vertical line at which vertical line interrupt is triggered.
(reserved)	15:11		
CRTC_CRNT_VLINE (R)	26:16	0x0	Current vertical line.
(reserved)	31:27		

CRTC_VLINE_CRNT_VLINE: Display Current Vertical Line

CRTC_GUI_TRIG_VLINE MMR:0x0218 MMR_1:0x0218 IND:0x0218 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_GUI_TRIG_VLINE_START	10:0	0x0	The START (upper in display, lower in memory) for the GUI_TRIG_VLINE compare. First line of display is line 0.
(reserved)	15:11		
CRTC_GUI_TRIG_VLINE_END	26:16	0x0	The END (lower in display, higher in memory) for the GUI_TRIG_VLINE compare. First line of display is line 0.
(reserved)	30:27		
CRTC_GUI_TRIG_VLINE (R)	31	0x0	This signal is active high when the raster is between the START and END. START <= raster <= END. 0=Current line not between VLINE start and end 1=Current line is between VLINE start and end, inclusive

CRTC_GUI_TRIG_VLINE: 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_OFFSET MMR:0x0224 MMR_1:0x0224 IND:0x0224 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_OFFSET	24:0	0x0	Primary graphics display base address in bytes. Updated for buffer flips and for virtual desktop movement. Must always point to the start of a character of display data (i.e. can not move horizontally by sub-character amount). Must pan by 16 pixels in 4bpp modes. In tiling mode, this field should be written with the start address of the display after tiling but before checkboarding. It means, in tiling mode the hardware won't convert this address to a linear equivalent (so this is not the 'virtual' tiled address but rather the 'real' physical address). But the programmer does not need to checkerboard this address, the hardware will do it. e.g.: surface offset is zero and the display will start at the beginning of line 3: CRTC_OFFSET = (surface base) + ((start line) * 0x40)= 0xC0 because each tile is 64 bytes wide. If the vertical offset exceeds the height of a tile (16 lines), then again the real address (before checkerboarding) of the start of the first line must be calculated. i.e. for tiled: CRTC_OFFSET = (surface base) + 16 * (((start line) DIV 16) * CRTC_PITCH) + 1K * ((x_start * bpp / 8) DIV 64) + 64 * (start line DIV 16) + (x_start * bpp /8) MOD 64. x_start is the pixel where the display starts in the image. NOTE: Bits 2:0 of this field are hardwired to ZERO
(reserved)	29:25		
CRTC_GUI_TRIG_OFFSET (R)	30	0x0	Indicates if visible buffer is last written, or still the previous one. This register 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 written is being displayed 1=Last CRTC_OFFSET written not yet displayed

CRTC_OFFSET_LOCK	31	0x0	Prevents hardware from internally updating the following fields until cleared: CRTC_OFFSET, CRTC_TILE_LINE. 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

CRTC_OFFSET: Graphics Base Address Offset

CRTC_OFFSET_CNTL MMR:0x0228 MMR_1:0x0228 IND:0x0228 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_TILE_LINE	4:0	0x0	When CRTC_TILE_EN = 1, this fields holds the 5 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. 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 16 lines high, but this register must contain ((start line) MOD 32) in order to do the checkerboarding correctly. Do not worry about what checkerboarding is, you shouldn't need to know.
(reserved)	7:5		

CRTC_TILE_ALIGN	10:8	0x0	Alignment of graphics display surface in tiled mode. Indicates memory alignment of the display surface (i.e. first tile), which is not the same as the one of CRTC_OFFSET if using virtual desktop. If the surface starts at an address multiple of 64 bytes, but not of 2K, set this to 0. If 2K but not 4K, 1 If 4K but not 8K, 2 If 8K but not 16K, 3 If 16K, 4 0=64 byte aligned surface 1=2k byte aligned surface 2=4k byte aligned surface 3=8k byte aligned surface 4=16k byte aligned surface
(reserved)	14:11		
CRTC_TILE_EN	15	0x0	Graphics display tiling enable. 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 offset 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.
(reserved)	29:17		
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 written is being displayed 1=Last CRTC_OFFSET written not yet displayed

CRTC_OFFSET_LOCK	31	0x0	Prevents hardware from internally updating the following fields until cleared: CRTC_OFFSET, CRTC_TILE_LINE. See CRTC_OFFSET register. 0=Unlock these regs 1=Lock'em

CRTC_OFFSET_CNTL: Graphics Display Address Generator Control

CRTC_PITCH MMR:0x022C MMR_1:0x022C IND:0x022C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_PITCH	9:0	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 (except for 24bpp, as above). This must be a multiple of 64 bytes (the basic tile width).
(reserved)	31:10		

CRTC_PITCH: Graphics Display Address Pitch

3.10 Memory Buffer Control

<No description>

MEM_ADDR_CONFIG MMR:0x0148 MMR_1:0x0148 IND:0x0148 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_ADDR_MAPPING	3:0	0x0	Row/Column/Banks address mapping of target memory. 0= 9 row bits x 8 col bits x 2 banks 1=10 row bits x 8 col bits x 2 banks 2=11 row bits x 8 col bits x 2 banks 3=12 row bits x 8 col bits x 2 banks : CS2 = A12 4=13 row bits x 8 col bits x 2 banks : CS2 = A12, CS3 = A13 8= 9 row bits x 8 col bits x 4 banks 9=10 row bits x 7 col bits x 4 banks 10=10 row bits x 8 col bits x 4 banks 11=11 row bits x 7 col bits x 4 banks : CS2 = A12 12=11 row bits x 8 col bits x 4 banks : CS2 = A12 13=12 row bits x 8 col bits x 4 banks : CS2 = A12, CS3 = A13
MEM_AP_MAPPING	6:4	0x0	Address bit used for auto-precharge function. 0=address bit 8 1=address bit 9 2=address bit 10 3=address bit 11 4=address bit 12
MEM_CS_4BANK_EN	7	0x0	<No Description> 0=Normal 1=Interleaving of banks across different memory groups
MEM_BUS_WIDTH	8	0x0	Memory Data bus width. 0=64 bits 1=128 bits
(reserved)	15:9		

MEM_CHECKBOARD	17:16	0x0	Address bit to 'twiddle' in order to get desired checkerboard pattern of tiled memory surfaces. 0=twiddle byte address bit 10 1=twiddle byte address bit 11 2=twiddle byte address bit 12 3=twiddle byte address bit 13
(reserved)	19:18		
MEM_BLKWR_MODE	21:20	0x0	Level of block write support of the memory. 0=Block write disabled 1=Block write disabled 2=Block write enabled without column byte mask 3=Block write enabled with column byte mask
(reserved)	31:22		

MEM_ADDR_CONFIG: Configuration of memory interface.

DDA_CONFIG MMR:0x02E0 MMR_1:0x02E0 IND:0x02E0 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DDA_XCLKS_PER_XFER	13:0	0x0	Amount of time in XCLKs that one transfer to the Display FIFO occupies
(reserved)	15:14		
DDA_PRECISION	19:16	0x0	Integer.Fraction precision point for DDA_XCLKS_PER_XFER DDA_ON DDA_OFF
DDA_LOOP_LATENCY	24:20	0x0	Display FIFO control parameter to reflect the number of XCLKs of latency required in the hardware.
(reserved)	31:25		

DDA_CONFIG: Contains DDA parameters that set the way data is fetch from memory to be displayed

DDA_ON_OFF MMR:0x02E4 MMR_1:0x02E4 IND:0x02E4 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DDA_OFF	15:0	0x0	The Display memory request off threshold time in terms of XCLKs

DDA_ON	31:16	0x0	The display memory request on threshold time in terms of XCLKs

DDA_ON_OFF: Indicates at what levels of the fifo to start and end fetching data

VGA_DDA_CONFIG MMR:0x02E8 MMR_1:0x02E8 IND:0x02E8 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VGA_DDA_XCLKS_PER_XFER	13:0	0x0	Amount of time in XCLKs that one transfer to the display FIFO occupies in VGA modes
(reserved)	19:14		
VGA_DDA_PREC_PCLKBY2	23:20	0x0	Integer.fraction precision point for: VGA_DDA_PREC_PCLK+1
VGA_DDA_PREC_PCLK	27:24	0x0	Integer.fraction precision point for: DDA_XCLKS_PER_XFER DDA_ON DDA_OFF
(reserved)	31:28		

VGA_DDA_CONFIG: <No Description>

VGA_DDA_ON_OFF MMR:0x02EC MMR_1:0x02EC IND:0x02EC [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VGA_DDA_OFF	15:0	0x0	The display memory request off threshold time in terms of XCLKs for VGA modes
VGA_DDA_ON	31:16	0x0	The display memory request on threshold time in terms of XCLKs for VGA modes

VGA_DDA_ON_OFF: <No Description>

3.11 General I/O Control

<No description>

3.12 Overscan

Display overscan is enabled if any of the overscan width values are non-zero. The left and right overscan widths are described in terms of pixels * 8 and the top and bottom overscan widths are described in terms of vertical lines. The overscan color is defined by an 8 bit index and a 24 bir color. In all display modes the 24 bit color will be used by the internal RAMDAC and displayed on the monitor attached to the RAGE128. Note this is always a true color that is not mapped through the palette. The 8 bit index color is used in 4 bpp and 8 bpp modes for data going out on the 8 bit feature connector. The receiving board is expected to index all 4 and 8 cpp data through it's own palette.

OVR_CLR MMR:0x0230 MMR_1:0x0230 IND:0x0230
[RW] 32 bits (access: 8/16/32)

Field Name	Bits	Default	Description
OVR_CLR_B	7:0	0x0	Blue overscan color.
OVR_CLR_G	15:8	0x0	Green overscan color.
OVR_CLR_R	23:16	0x0	Red overscan color.
(reserved)	31:24		

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

OVR_WID_LEFT_RIGHT MMR:0x0234 MMR_1:0x0234 IND:0x0234
[RW] 32 bits (access: 8/16/32)

Field Name	Bits	Default	Description
OVR_WID_RIGHT	5:0	0x0	Right overscan width (in pixels * 8).
(reserved)	15:6		
OVR_WID_LEFT	21:16	0x0	Left overscan width (in pixels * 8).
(reserved)	31:22		

OVR_WID_LEFT_RIGHT: Overscan border left/right width control.

OVR_WID_TOP_BOTTOM MMR:0x0238 MMR_1:0x0238 IND:0x0238
[RW] 32 bits (access: 8/16/32)

Field Name	Bits	Default	Description
OVR_WID_BOTTOM	8:0	0x0	Bottom overscan width (in scan lines).
(reserved)	15:9		

OVR_WID_TOP	24:16	0x0	Top overscan width (in scan lines).
(reserved)	31:25		

OVR_WID_TOP_BOTTOM: Overscan border top/bottom width control.

3.13 Hardware Cursor

<No description>

CUR_OFFSET MMR:0x0260 MMR_1:0x0260 IND:0x0260 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CUR_OFFSET	24:0	0x0	Hardware cursor address offset. Must be in the frame buffer, and be 16 byte (128 bit) aligned. This value is adjusted to move the cursor off the top edge of the display. See the CUR_VERT_OFF description. NOTE: Bits 3:0 of this field are hardwired to ZERO
(reserved)	30:25		
CUR_LOCK	31	0x0	Locks the 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

CUR_OFFSET: Location of the hardware cursor image.

CUR_HORZ_VERT_POSN MMR:0x0264 MMR_1:0x0264 IND:0x0264 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CUR_VERT_POSN	10:0	0x0	Cursor vertical position. To move the cursor off the top edge set CUR_VERT_POSN=0 and see the CUR_VERT_OFF description.
(reserved)	15:11		
CUR_HORZ_POSN	26:16	0x0	Cursor horizontal position. To move the cursor off the left edge set CUR_HORZ_POSN=0 and see the CUR_HORZ_OFF description.
(reserved)	30:27		

CUR_LOCK	31	0x0	Locks the 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

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

CUR_HORZ_VERT_OFF MMR:0x0268 MMR_1:0x0268 IND:0x0268 [RW] 32 bits (access: 8/16/32)			
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.
(reserved)	15:6		
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.
(reserved)	30:22		
CUR_LOCK	31	0x0	Locks the 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

CUR_HORZ_VERT_OFF: 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 MMR:0x026C MMR_1:0x026C IND:0x026C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CUR_CLR0_B	7:0	0x0	Blue cursor color 0.
CUR_CLR0_G	15:8	0x0	Green cursor color 0.
CUR_CLR0_R	23:16	0x0	Red cursor color 0.
(reserved)	31:24		

CUR_CLR0: Hardware cursor color 0. Always 24bpp, independent of graphics mode.

CUR_CLR1 MMR:0x0270 MMR_1:0x0270 IND:0x0270 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CUR_CLR1_B	7:0	0x0	Blue cursor color 1.
CUR_CLR1_G	15:8	0x0	Green cursor color 1.
CUR_CLR1_R	23:16	0x0	Red cursor color 1.
(reserved)	31:24		

CUR_CLR1: Hardware cursor color 1. Always 24bpp, independent of graphics mode.

3.14 GenLocking Register

<No description>

3.15 Scratch Pad

<No description>

BIOS_0_SCRATCH MMR:0x0010 MMR_1:0x0010 IOR:0x0010 IND:0x0010 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BIOS_0_SCRATCH	31:0	0x0	Scratch memory for use by video BIOS.

BIOS_0_SCRATCH: BIOS Scratch 0

BIOS_1_SCRATCH MMR:0x0014 MMR_1:0x0014 IOR:0x0014 IND:0x0014 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BIOS_1_SCRATCH	31:0	0x0	Scratch memory for use by video BIOS.

BIOS_1_SCRATCH: BIOS Scratch 1

BIOS_2_SCRATCH MMR:0x0018 MMR_1:0x0018 IOR:0x0018 IND:0x0018 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BIOS_2_SCRATCH	31:0	0x0	Scratch memory for use by video BIOS.

BIOS_2_SCRATCH: BIOS Scratch 2

BIOS_3_SCRATCH MMR:0x001C MMR_1:0x001C IOR:0x001C IND:0x001C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BIOS_3_SCRATCH	31:0	0x0	Scratch memory for use by video BIOS.

BIOS_3_SCRATCH: BIOS Scratch 3

3.16 Clock Control

<No description>

CLOCK_CNTL_INDEX MMR:0x0008 MMR_1:0x0008 IOR:0x0008 IND:0x0008 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PLL_ADDR	4:0	0x0	<No Description>
(reserved)	6:5		
PLL_WR_EN	7	0x0	Enables writes to the Clock Control Registers through the CLOCK_CNTL_DATA index register 0=Disable 1=Enable
PPLL_DIV_SEL	9:8	0x0	Selects which of the 3 PLL Clock dividers generates the source for the pixel clock 0=PPLL_DIV0 1=PPLL_DIV1 2=PPLL_DIV2 3=PPLL_DIV3
(reserved)	31:10		

CLOCK_CNTL_INDEX: <No Description>

CLOCK_CNTL_DATA MMR:0x000C MMR_1:0x000C IOR:0x000C IND:0x000C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PLL_DATA	31:0	0x0	<No Description>

CLOCK_CNTL_DATA: <No Description>

3.17 PLL Registers

<No description>

CLK_PIN_CNTL PLL:0x0001 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
OSC_EN	0	0x1	<No Description> 0=Disable 1=Enable
DCLK_TRI_STATE	1	0x1	<No Description> 0=OutputEn 1=TriState
XTL_LOW_GAIN	2	0x1	<No Description> 0=High GAIN 1=Low GAIN
(reserved)	3		
HCLK0_OUT_EN	4	0x1	Tristate/Output selection for clock pin HCLK0. 0=CLK0 pin tristated 1=CLK0 pin output enabled
HCLK0b_OUT_EN	5	0x1	Tristate/Output selection for clock pin HCLK0b. 0=CLK0b pin tristated 1=CLK0b pin output enabled
HCLK1_OUT_EN	6	0x1	Tristate/Output selection for clock pin HCLK1. 0=CLK1 pin tristated 1=CLK1 pin output enabled
HCLK1b_OUT_EN	7	0x1	Tristate/Output selection for clock pin HCLK1b. 0=CLK1b pin tristated 1=CLK1b pin output enabled
(reserved)	31:8		

CLK_PIN_CNTL: Control of clock pins.

PPLL_CNTL PLL:0x0002 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PPLL_RESET	0	0x1	<No Description> 0=Not Reset 1=Reset
PPLL_SLEEP	1	0x1	<No Description> 1=Powerdown
(reserved)	7:2		
PPLL_PC_GAIN	10:8	0x4	<No Description>
PPLL_VC_GAIN	12:11	0x1	<No Description>
PPLL_DCYC	14:13	0x2	PPLL_DCYC(1) controls IPPDC1; PPLL_DCYC(0) controls IPPDC0; The PPLL_DCYC and PPLL_RANGE together control the duty cycle.
PPLL_RANGE	15	0x1	This controls IPPDC2 of the PCLK PLL
PPLL_ATOMIC_UPDATE_EN	16	0x0	<No Description> 0=Atomic Update Disabled 1=Atomic Update Enabled
PPLL_VGA_ATOMIC_UPDATE_EN	17	0x0	<No Description> 0=VGA Atomic Update Disabled 1=VGA Atomic Update Enabled
PPLL_ATOMIC_UPDATE_SYNC	18	0x0	<No Description> 0=Update ASAP 1=Update in VSYNC
(reserved)	31:19		

PPLL_CNTL: <No Description>

PPLL_REF_DIV PLL:0x0003 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PPLL_REF_DIV	9:0	0x0	Reference divider for PPLL. This is 'M' in the PLL frequency equation: PPllClk = N*PPLL_REF/M Reference divider must be >= 2, otherwise divider stops operating. Upper limit determined by PPLL_REF/M must be >= 200kHz. In general, M is set as large as possible to satisfy the last restriction.
(reserved)	14:10		
PPLL_ATOMIC_UPDATE_R (R)	15	0x0	Indicates progress of last request for update to PPLL_REF_DIV and/or PPLL_FBx_DIV. 0=Update done 1=Update Pending
PPLL_ATOMIC_UPDATE_W (W)	15	0x0	Used to update new settings into PPLL reference and feedback dividers for GEN-locking. Should be used when setting new PPLL_REF_DIV or PPLL_FBx_DIV, or when changing PPLL_DIV_SEL or VGA_CKSEL to use different PPLL_FBx_DIV. Not required if not enabled in PPLL_ATOMIC_UPDATE_EN or PPLL_VGA_ATOMIC_UPDATE_EN. 0=No Update 1=Update
PPLL_REF_DIV_SRC	17:16	0x0	Selects the input to the PPLL reference divider. 0=PPLL_REF = XTALIN 1=PPLL_REF = MPllClk/2 2=PPLL_REF = XPllClk/2
(reserved)	31:18		

PPLL_REF_DIV: Pixel clock PLL reference divider controls

PPLL_DIV_0 PLL:0x0004 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB0_DIV	10:0	0x0	Feedback divider for PPLL when clock select is 0. This is 'N' in the PLL frequency equation: PPllClk = (N*PPLL_REF)/M Feedback divider must be >= 4, otherwise divider stops operating. Legal range for PPllClk frequency is 125 MHz to 250 MHz. M and N must be chosen to satisfy this restriction, and the upper bound limit on M stated for PPLL_REF_DIV.
(reserved)	14:11		
PPLL_ATOMIC_UPDATE_R (R)	15	0x0	Indicates progress of last request for update to PPLL_REF_DIV and/or PPLL_FBx_DIV. 0=Update done 1=Update Pending
PPLL_ATOMIC_UPDATE_W (W)	15	0x0	Used to update new settings into PPLL reference and feedback dividers for GEN-locking. Should be used when setting new PPLL_REF_DIV or PPLL_FBx_DIV, or when changing PPLL_DIV_SEL or VGA_CKSEL to use different PPLL_FBx_DIV. Not required if not enabled in PPLL_ATOMIC_UPDATE_EN or PPLL_VGA_ATOMIC_UPDATE_EN. 0=No Update 1=Update
PPLL_POST0_DIV	18:16	0x0	Selects PPLL post-divider for when PPLL clock select is 0. If doing TV output, then must be set to /2 or /3 as indicated. 0=VCLK = VCLK_SRC 1=VCLK = VCLK_SRC/2, required for TV out 565 2=VCLK = VCLK_SRC/4 3=VCLK = VCLK_SRC/8 4=VCLK = VCLK_SRC/3, required for TV out 888 5=reserved 6=VCLK = VCLK_SRC/6 7=VCLK = VCLK_SRC/12
(reserved)	31:19		

PPLL_DIV_0: PPLL feedback and post divider settings for when PPLL clock select is 0.

PPLL clock select is VGA_CKSEL@GENMO in VGA modes or PPLL_DIV_SEL@CLOCK_CNTL_INDEX in non-VGA modes.
CRTC_EXT_DISP_EN@CRTC_GEN_CNTL= 0 indicates VGA mode.

PPLL_DIV_1 PLL:0x0005
[RW] 32 bits (access: 8/16/32)

Field Name	Bits	Default	Description
PPLL_FB1_DIV	10:0	0x0	Feedback divider for PPLL when clock select is 1. This is 'N' in the PLL frequency equation: PPllClk = (N*PPLL_REF)/M Feedback divider must be >= 4, otherwise divider stops operating. Legal range for PPllClk frequency is 125 MHz to 250 MHz. M and N must be chosen to satisfy this restriction, and the upper bound limit on M stated for PPLL_REF_DIV.
(reserved)	14:11		
PPLL_ATOMIC_UPDATE_R (R)	15	0x0	Indicates progress of last request for update to PPLL_REF_DIV and/or PPLL_FBx_DIV. 0=Update done 1=Update Pending
PPLL_ATOMIC_UPDATE_W (W)	15	0x0	Used to update new settings into PPLL reference and feedback dividers for GEN-locking. Should be used when setting new PPLL_REF_DIV or PPLL_FBx_DIV, or when changing PPLL_DIV_SEL or VGA_CKSEL to use different PPLL_FBx_DIV. Not required if not enabled in PPLL_ATOMIC_UPDATE_EN or PPLL_VGA_ATOMIC_UPDATE_EN. 0=No Update 1=Update

PPLL_POST1_DIV	18:16	0x0	Selects PPLL post-divider for when PPLL clock select is 1. If doing TV output, then must be set to /2 or /3 as indicated. 0=VCLK = VCLK_SRC 1=VCLK = VCLK_SRC/2, required for TV out 565 2=VCLK = VCLK_SRC/4 3=VCLK = VCLK_SRC/8 4=VCLK = VCLK_SRC/3, required for TV out 888 5=reserved 6=VCLK = VCLK_SRC/6 7=VCLK = VCLK_SRC/12
(reserved)	31:19		

PPLL_DIV_1: PPLL feedback and post divider settings for when PPLL clock select is 1.
PPLL clock select is VGA_CKSEL@GENMO in VGA modes or PPLL_DIV_SEL@CLOCK_CNTL_INDEX in non-VGA modes.
CRTC_EXT_DISP_EN@CRTC_GEN_CNTL= 0 indicates VGA mode.

PPLL_DIV_2 PLL:0x0006
[RW] 32 bits (access: 8/16/32)

Field Name	Bits	Default	Description
PPLL_FB2_DIV	10:0	0x0	Feedback divider for PPLL when clock select is 2. This is 'N' in the PLL frequency equation: PPllClk = (N*PPLL_REF)/M Feedback divider must be >= 4, otherwise divider stops operating. Legal range for PPllClk frequency is 125 MHz to 250 MHz. M and N must be chosen to satisfy this restriction, and the upper bound limit on M stated for PPLL_REF_DIV.
(reserved)	14:11		
PPLL_ATOMIC_UPDATE_R (R)	15	0x0	Indicates progress of last request for update to PPLL_REF_DIV and/or PPLL_FBx_DIV. 0=Update done 1=Update Pending

PPLL_ATOMIC_UPDATE_W (W)	15	0x0	Used to update new settings into PPLL reference and feedback dividers for GEN-locking. Should be used when setting new PPLL_REF_DIV or PPLL_FBx_DIV, or when changing PPLL_DIV_SEL or VGA_CKSEL to use different PPLL_FBx_DIV. Not required if not enabled in PPLL_ATOMIC_UPDATE_EN or PPLL_VGA_ATOMIC_UPDATE_EN. 0=No Update 1=Update
PPLL_POST2_DIV	18:16	0x0	Selects PPLL post-divider for when PPLL clock select is 2. If doing TV output, then must be set to /2 or /3 as indicated. 0=VCLK = VCLK_SRC 1=VCLK = VCLK_SRC/2, required for TV out 565 2=VCLK = VCLK_SRC/4 3=VCLK = VCLK_SRC/8 4=VCLK = VCLK_SRC/3, required for TV out 888 5=reserved 6=VCLK = VCLK_SRC/6 7=VCLK = VCLK_SRC/12
(reserved)	31:19		

PPLL_DIV_2: PPLL feedback and post divider settings for when PPLL clock select is 2.

PPLL clock select is VGA_CKSEL@GENMO in VGA modes or PPLL_DIV_SEL@CLOCK_CNTL_INDEX in non-VGA modes.

CRTC_EXT_DISP_EN@CRTC_GEN_CNTL= 0 indicates VGA mode.

PPLL_DIV_3 PLL:0x0007 **[RW] 32 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
PPLL_FB3_DIV	10:0	0x0	Feedback divider for PPLL when clock select is 3. This is 'N' in the PLL frequency equation: PPllClk = (N*PPLL_REF)/M Feedback divider must be >= 4, otherwise divider stops operating. Legal range for PPllClk frequency is 125 MHz to 250 MHz. M and N must be chosen to satisfy this restriction, and the upper bound limit on M stated for PPLL_REF_DIV.
(reserved)	14:11		

PPLL_ATOMIC_UPDATE_R (R)	15	0x0	Indicates progress of last request for update to PPLL_REF_DIV and/or PPLL_FBx_DIV. 0=Update done 1=Update Pending
PPLL_ATOMIC_UPDATE_W (W)	15	0x0	Used to update new settings into PPLL reference and feedback dividers for GEN-locking. Should be used when setting new PPLL_REF_DIV or PPLL_FBx_DIV, or when changing PPLL_DIV_SEL or VGA_CKSEL to use different PPLL_FBx_DIV. Not required if not enabled in PPLL_ATOMIC_UPDATE_EN or PPLL_VGA_ATOMIC_UPDATE_EN. 0=No Update 1=Update
PPLL_POST3_DIV	18:16	0x0	Selects PPLL post-divider for when PPLL clock select is 3. If doing TV output, then must be set to /2 or /3 as indicated. 0=VCLK = VCLK_SRC 1=VCLK = VCLK_SRC/2, required for TV out 565 2=VCLK = VCLK_SRC/4 3=VCLK = VCLK_SRC/8 4=VCLK = VCLK_SRC/3, required for TV out 888 5=reserved 6=VCLK = VCLK_SRC/6 7=VCLK = VCLK_SRC/12
(reserved)	31:19		

PPLL_DIV_3: PPLL feedback and post divider settings for when PPLL clock select is 3.

PPLL clock select is VGA_CKSEL@GENMO in VGA modes or PPLL_DIV_SEL@CLOCK_CNTL_INDEX in non-VGA modes.

CRTC_EXT_DISP_EN@CRTC_GEN_CNTL= 0 indicates VGA mode.

VCLK_ECP_CNTL PLL:0x0008 **[RW] 32 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**

VCLK_SRC_SEL	1:0	0x0	Selects source of VCLK. If set to BYTE_CLK, then see BYTE_CLK_POST_DIV below to select the VCLK source. Both the clock source you are switching to and from must be running, or the switch will not occur. 0=VCLK_SRC = PCICLK (input pin) 1=VCLK_SRC = PCLK (input pin) 2=VCLK_SRC = BYTE_CLK (see below) 3=VCLK_SRC = PPllClk
(reserved)	3:2		
VCLK_INVERT	4	0x0	Used to invert VCLK 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
(reserved)	7:5		
ECP_DIV	9:8	0x0	Determines horizontal replication factor for back-end overlay/scalar output. ECP can not exceed 125 MHz. If VCLK <= 125 MHz, then set ECP = VCLK. If 125 MHz < VCLK <= 250 MHz, then set ECP = VCLK/2. etc. Overlay/scalar will produce one scaled output pixel for each period of ECP. 0=ECP = VCLK 1=ECP = VCLK/2 2=ECP = VCLK/4
(reserved)	15:10		

BYTE_CLK_POST_DIV	17:16	0x0	Selects the source of BYTE_CLK when TV output port is enabled. Select BYTCLK input pin when ImpacTV/Rage Theater are to be the dot clock source. Select the apropriate post-divider when Rage graphics chip is to supply the dot clock source. In this case, this post-divider determines the speed of BYTE_CLK to the ImpacTV/Rage Theater, and the PPLL_POSTx_DIV determines if 16 or 24 data bits per pixel are sent over. Don't care if VCLK_SRC_SEL <> 10. 0=BYTE_CLK = BYTCLK (input pin) 1=BYTE_CLK = PPllCLk/2 2=BYTE_CLK = PPllCLk/3 3=BYTE_CLK = PPllCLk/4
ECP_FORCE_ON	18	0x0	Controls the dynamic clock control for the back-end overlay/scalar. Set to low for power reduction. 0=SCALAR ACTIVITY 1=CONTINUOUS
(reserved)	19		
BYTE_CLK_OUT_EN	20	0x0	Controls the function of the BYTCLK pin. When BYTE_CLK_POST_DIV = 00 (BYTCLK input), then this bit should be low (tri state). For other settings of BYTE_CLK_POST_DIV, this bit should be high to drive the clock to the TV encoder chip. 0=Tri State BYTCLK output 1=Enable BYTCLK output = BYTE_CLK
(reserved)	23:21		
BYTE_CLK_SKEW	26:24	0x0	Selects phase of internally generated BYTE_CLK to VCLK. Don't care if BYTE_CLK_POST_DIV = 00 (external input). Used to do alignment of TV out data with clock when Rage graphics chip is generating the dot clock for TV output. Selects phase in 1/2 PPllClk increments. Valid range depends on BYTE_CLK_POST_DIV setting, and may not exceed (2*byte clock post divider)-1. e.g. for byte clock post divider of 3 (BYTE_CLK_POST_DIV=10), then (2*3)-1=5, so BYTE_CLK_SKEW has range 0 to 5.
(reserved)	31:27		

VCLK_ECP_CNTL: General controls for the display clocks.

VCLK is the pixel, or dot, clock.
ECP is the overlay/scalar clock.

HTOTAL_CNTL PLL:0x0009 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
HTOT_PIX_SLIP	3:0	0x0	Pixel accurate control of HTOTAL. Selects the extra number of pixels to add to each display line. Valid range is 0 to 7. For VGA modes with SEQ_PCLKBY2 = 1 each increment adds two pixels to the line total. For 9-dot VGA text modes, it is not possible to add 8/9ths of a character extra to the HTOTAL value.
(reserved)	7:4		
HTOT_VCLK_SLIP	11:8	0x0	The highest bit is to enable the slipping capability of VCLK through PDATA0. 0=highest bit is to enable the logic to synchronize between master/slave chip
(reserved)	15:12		
HTOT_PPLL_SLIP	18:16	0x0	Finest adjustment control. This selects the number of VCO phase slips to do in the PLL at every HSYNC. Each VCO phase slip is equal to 0.2 of a PLLVCLK period.
(reserved)	23:19		
HTOT_CNTL_EDGE	24	0x0	Select which HTOTAL edge the correction is done on: 0 = rising edge of HSYNC 1 = falling edge of HSYNC
(reserved)	27:25		
HTOT_CNTL_VGA_EN	28	0x0	Selects whether the HTOT_PPLL_SLIP & HTOT_VCLK_SLIP are enable for VGA display modes. 0 = not enabled for VGA modes 1 = enabled for VGA modes
(reserved)	31:29		

HTOTAL_CNTL: 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).

X_MPLL_REF_FB_DIV PLL:0x000A [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
X_MPLL_REF_DIV	7:0	0x0	Reference divider for both MPLL and XPLL. This is 'M' in the PLL frequency equation: PllClk = 2*N*Xtalin/M Reference divider must be >= 2, otherwise divider stops operating. Upper limit determined by Xtalin/M must be >= 400kHz. In general, M is set as large as possible to satisfy the last restriction.
XPLL_FB_DIV	15:8	0x0	Feedback divider for XPLL. This is 'N' in the PLL frequency equation: XPllClk = 2*N*Xtalin/M Feedback divider must be >= 2, otherwise divider stops operating. Legal range for XPllClk frequency is 125 MHz to 250 MHz. M and N must be chosen to satisfy this restriction, and the upper bound limit on M stated above.
MPLL_FB_DIV	23:16	0x0	Feedback divider for MPLL. This is 'N' in the PLL frequency equation: MPllClk = 2*N*Xtalin/M Feedback divider must be >= 2, otherwise divider stops operating. Legal range for MPllClk frequency is 125 MHz to 250 MHz. M and N must be chosen to satisfy this restriction, and the upper bound limit on M stated above.
(reserved)	31:24		

X_MPLL_REF_FB_DIV: PLL reference and feedback settings for MPLL and XPLL

XPLL_CNTL PLL:0x000B [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
XPLL_RESET	0	0x1	<No Description> 0=Not Reset 1=Reset

XPLL_SLEEP	1	0x1	<No Description> 1=Powerdown
(reserved)	7:2		
XPLL_PC_GAIN	10:8	0x4	XPLL charge pump gain setting
XPLL_VC_GAIN	12:11	0x1	XPLL VCGEN gain setting
XPLL_DCYC	14:13	0x2	XPLL_DCYC(1) controls IXPDC1; XPLL_DCYC(0) controls IXPDC0; The XPLL_DCYC and XPLL_RANGE togehter controls the duty cycle
XPLL_RANGE	15	0x1	This bit controls IXPDC2
(reserved)	31:16		

XPLL_CNTL: PLL macro controls for XPLL

XDLL_CNTL PLL:0x000C **[RW] 32 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
XDLL0_SLEEP	0	0x1	Sleep mode for DLL0. 0=Enabled 1=PowerDown
XDLL0_RESET	1	0x1	Reset for DLL0. 0=Enabled 1=Reset
XDLL0_RANGE	3:2	0x2	Frequency range for DLL0. 2= 80 MHz to 110 MHz 3=110 MHz to 150 MHz

XDLL0_REF_SEL	5:4	0x0	Reference select for DLL0. 0=XCLK 1=HCLK0 pad 2=not YCLK
XDLL0_FB_SEL	7:6	0x0	Feedback select for DLL0. 0=HCLK0 pad 1=QS0 pad 2=internal feedback
XDLL0_REF_SKEW	10:8	0x0	Skew of reference signal selected by XDLL0_REF_SEL.
(reserved)	11		
XDLL0_FB_SKEW	14:12	0x0	Skew of feedback signal selected by XDLL0_FB_SEL.
(reserved)	15		
XDLL1_SLEEP	16	0x1	Sleep mode for DLL1. 0=Enabled 1=PowerDown
XDLL1_RESET	17	0x1	Reset for DLl1. 0=Enabled 1=Reset
XDLL1_RANGE	19:18	0x2	Frequency range of DLL1. 2= 80 MHz to 110 MHz 3=110 MHz to 150 MHz
XDLL1_REF_SEL	21:20	0x0	Reference select for DLL1. 0=XCLK 1=HCLK1 pad 2=YCLKb
XDLL1_FB_SEL	23:22	0x0	Feedback select for DLL1. 0=HCLK1 pad 1=QS1 pad 2=internal feedback

XDLL1_REF_SKEW	26:24	0x0	Skew of reference signal selected by XDLL1_REF_SEL.
(reserved)	27		
XDLL1_FB_SKEW	30:28	0x0	Skew of feedback signal selected by XDLL1_FB_SEL.
(reserved)	31		

XDLL_CNTL: DLL Control Register.

XCLK_CNTL PLL:0x000D **[RW] 32 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
XCLK_SRC_SEL	3:0	0x0	Selection for XCLK. 0=XCLK = not PCICLK 1=XCLK = XPllClk 2=XCLK = XPllClk/2 3=XCLK = XPllClk/4 4=XCLK = XPllClk/8 5=XCLK = HCLK0 (direct) 6=XCLK = HCLK1 (direct) 7=XCLK = XDLL0CLK 8=XCLK = disabled/stopped
YCLK_SRC_SEL	7:4	0x0	Selection for YCLK. 0=YCLK = not PCICLK 1=YCLK = XPllClk 2=YCLK = XPllClk/2 3=YCLK = XPllClk/4 4=YCLK = XPllClk/8 5=YCLK = HCLK0 (direct) 6=YCLK = HCLK1 (direct) 7=YCLK = XDLL0CLK 8=YCLK = disabled/stopped
HCLK0_SEL	10:8	0x0	Selection for HCLK0 pin. 0=HCLK0 = XCLK 1=HCLK0 = not XCLK 2=HCLK0 = not YCLK 3=HCLK0 = YCLK/2 4=HCLK0 = XDLL0CLK

HCLK0_REC	11	0x0	Receiver mode for HCLK0 pin. 0=hysteresis receiver 1=differential receiver
HCLK1_SEL	14:12	0x0	Selection for HCLK1 pin. 0=HCLK1 = XCLK 1=HCLK1 = not XCLK 2=HCLK1 = not YCLK 3=HCLK1 = YCLK/2 4=HCLK1 = XDLL1CLK
HCLK1_REC	15	0x0	Receiver mode for HCLK1 pin. 0=hysteresis receiver 1=differential receiver
(reserved)	31:16		

XCLK_CNTL: Clock control register for XCLK clock family.

MPLL_CNTL PLL:0x000E [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MPLL_SLEEP	0	0x1	<No Description> 1=Powerdown
MPLL_RESET	1	0x1	<No Description> 0=Not Reset 1=Reset
(reserved)	7:2		
MPLL_PC_GAIN	10:8	0x4	<No Description>
MPLL_VC_GAIN	12:11	0x1	<No Description>

MPLL_DCYC	14:13	0x2	MPLL_DCYC(1) controls IMPDC1; MPLL_DCYC(0) controls IMPDC0; The MPLL_DCYC and MPLL_RANGE togehter controls the duty cycle
MPLL_RANGE	15	0x1	this bit controls IMPDC2;
(reserved)	31:16		

MPLL_CNTL: <No Description>

MCLK_CNTL PLL:0x000F [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MCLK_SRC_SEL	2:0	0x0	MCLK (Main Clock) source selection. Must always switch from a running clock to a running clock, or hang can occur. 0=PCICLK 1=MPllClk/1 2=MPllClk/2 3=MPllClk/4 4=MPllClk/8 5=XCLK 6=reserved 7=XTALIN
(reserved)	15:3		
FORCE_GCP	16	0x0	Controls the dynamic clocking for the 2D engine. Set to low for power reduction. 0=Dynamic 1=ForceOn
FORCE_PIPE3D_CP	17	0x0	Controls the dynamic clocking for the 3D engine. Set to low for power reduction. 0=Dynamic 1=ForceOn

FORCE_RCP	18	0x0	Controls the dynamic clocking for the internal registers. Set to low for power reduction. 0=Dynamic 1=ForceOn
(reserved)	31:19		

MCLK_CNTL: General controls for the 'Engine' clock. Also known as the 'Main' clock.

AGP_PLL_CNTL PLL:0x0010 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
APLL_SLEEP	0	0x0	<No Description>
APLL_RESET	1	0x0	<No Description>
(reserved)	7:2		
APLL_XSEL	9:8	0x0	<No Description>
(reserved)	15:10		
APLL_X1_CLK_SKEW	18:16	0x7	This controls the skew of ISB inside the AGP_PLL
(reserved)	19		
APLL_X2_CLK_SKEW	22:20	0x7	This controls the skew of ISC inside the AGP_PLL
(reserved)	23		
APLL_TST_EN	24	0x0	<No Description>
APLL_PUMP_GAIN	26:25	0x1	<No Description>
APLL_VCO_GAIN_LOW	27	0x1	<No Description>
APLL_X4_CLK_SKEW	30:28	0x7	This controls the skew of ISD inside the AGP_PLL
(reserved)	31		

AGP_PLL_CNTL: <No Description>

FCP_CNTL PLL:0x0012 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

FCP0_SRC_SEL	2:0	0x4	<No Description> 0=PCICLK 1=PCLK 2=PCLKb 3=HREF 4=GND 5=HREFb
(reserved)	7:3		
FCP1_SRC_SEL	10:8	0x4	<No Description> 0=PCICLK 1=PCLK 2=PCLKb 3=HREF 4=GND 5=HREFb
(reserved)	31:11		

FCP_CNTL: <No Description>

PLL_TEST_CNTL PLL:0x0013 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	7:0		
TST_DIVIDERS	8	0x0	<No Description>
PLL_MASK_READ_B	9	0x1	<No Description>
(reserved)	15:10		
ANALOG_MON	19:16	0x0	<No Description>
(reserved)	23:20		
TEST_COUNT	31:24	0x0	<No Description>

PLL_TEST_CNTL: <No Description>

3.18 Bus Control

<No description>

BUS_CNTL MMR:0x0030 MMR_1:0x0030 IOR:0x0030 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
BUS_DBL_RESYNC	0	0x1	<No Description> 0=Normal 1=Add extra resynchronizing clock
BUS_MSTR_RESET (W)	1	0x0	<No Description> 0=Normal 1=Reset
BUS_FLUSH_BUF (W)	2	0x0	<No Description> 0=Normal 1=Flush
BUS_STOP_REQ_DIS	3	0x0	<No Description> 0=Normal 1=Disable
BUS_QUE_ACTIVE_DIS	4	0x0	<No Description> 0=Normal 1=Disable
BUS_ROTATION_DIS	5	0x0	<No Description> 0=Enable 1=Disable
BUS_MASTER_DIS	6	0x1	<No Description> 0=Enable 1=Disable

BIOS_ROM_WRT_EN	7	0x0	<No Description> 0=Disable 1=Enable
BUS_OS_READ_REQ	11:8	0xf	<No Description>
BIOS_DIS_ROM	12	0x0	<No Description> 0=Enable 1=Disable
BUS_PCI_READ_RETRY_EN	13	0x0	<No Description> 0=Normal 1=Enable
BUS_AGP_AD_STEPPING_EN	14	0x1	<No Description> 0=No stepping in AGP 1=AD Stepping in AGP and PCI
BUS_PCI_WRT_RETRY_EN	15	0x0	<No Description> 0=Normal 1=Enable
BUS_RETRY_WS	19:16	0xf	<No Description>
BUS_MSTR_RD_MULT	20	0x0	<No Description> 0=Read line 1=Read multiple
BUS_MSTR_RD_LINE	21	0x0	<No Description> 0=Read multiple 1=Read line
BUS_SUSPEND	22	0x0	<No Description> 0=Resume BM transfer 1=Suspend BM transfer

LAT_16X	23	0x0	<No Description> 0=1X 1=16X
BUS_RD_DISCARD_EN	24	0x0	<No Description> 0=Disable 1=Enable
BUS_RD_ABORT_EN	25	0x0	<No Description> 0=Disable 1=Enable
BUS_MSTR_WS	26	0x0	<No Description> 0=8 wait states 1=32 wait states
BUS_PARKING_DIS	27	0x1	<No Description> 0=Enable 1=Disable
BUS_MSTR_DISCONNECT_EN	28	0x0	<No Description> 0=Disable 1=Enable
BUS_CONTINUE_EN	29	0x0	<No Description> 0=Disable 1=Enable
BUS_READ_BURST	30	0x0	<No Description> 0=Disable 1=Enable
BUS_RDY_READ_DLY	31	0x1	<No Description> 0=no RDY delay 1=RDY delayed 1 mem clk

BUS_CNTL: <No Description>

3.19 Memory

<No description>

MEM_VGA_WP_SEL MMR:0x0038 MMR_1:0x0038 IOR:0x0038 IND:0x0038 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_VGA_WPS0	9:0	0x0	<No Description>
(reserved)	15:10		
MEM_VGA_WPS1	25:16	0x0	<No Description>
(reserved)	31:26		

MEM_VGA_WP_SEL: <No Description>

MEM_VGA_RP_SEL MMR:0x003C MMR_1:0x003C IOR:0x003C IND:0x003C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_VGA_RPS0	9:0	0x0	<No Description>
(reserved)	15:10		
MEM_VGA_RPS1	25:16	0x0	<No Description>
(reserved)	31:26		

MEM_VGA_RP_SEL: <No Description>

MEM_CNTL MMR:0x0140 MMR_1:0x0140 IND:0x0140 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_CFG_TYPE	1:0	0x0	Configuration type of memory interface. 0=SDR SGRAM (1:1) 1=SDR SGRAM (2:1) 2=DDR SGRAM
(reserved)	2		

MEM_BW_COL	3	0x0	Number of columns written by block write command. 0=8 columns 1=16 columns
MEM_ERST_CNTL	5:4	0x0	Delay of internal ERST signal after read command. Only relevant when memory configuration type is SDR2:1 or DDR. 0=(CL-1) clocks 1=(CL-1/2) clocks 2=CL clocks 3=Always enabled
MEM_DREN_CNTL	7:6	0x0	Delay of internal DRAN signal after read command. Only relevant when memory configuration type is SDR2:1 or DDR. 0=(CL-1) clocks 1=(CL-1/2) clocks 2=CL clocks 3=Always enabled
MEM_LATENCY	10:8	0x3	Memory read data latching delay from read command. 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
(reserved)	11		
MEM_WR_LATENCY	13:12	0x0	Latency of write data after write command. 0=0 clocks 1=1/2 clocks 2=1 clock
MEM_WDOE_CNTL	15:14	0x0	Control of when to drive write data bus relative to write command. 0=1 clock before 1=1/2 clock before 2=0 clocks before 3=1/2 clock after

MEM_OPER_MODE	17:16	0x0	Operating mode of the sequencer. 0=Normal 1=Page Hiding Disabled
MEM_INIT_DDR_SEL	18	0x0	<No Description> 0=Read DDR data zero first 1=Read DDR data one first
(reserved)	19		
MEM_CTLR_STATUS (R)	20	0x0	Memory Controller busy indicator. 0=Idle 1=Busy
MEM_SEQNCR_STATUS (R)	21	0x0	Memory Controller's sequencer busy indicator. 0=Idle 1=Busy
MEM_ARBITER_STATUS (R)	22	0x0	Memory Controller's arbiter busy indicator. 0=Idle 1=Busy
MEM_REQ_LOCK	23	0x0	Locks out new client requests from being accepted by the memory controller. 0=Unlocked 1=Lock Out Requestors
MEM_EXTND_ERST	24	0x0	Extend internal ERST signal an additional clock cycle. Only relevant when memory configuration type is SDR2:1 or DDR. 0=No Extension 1=Extend
MEM_EXTND_DREN	25	0x0	Extend internal DREN signal an additional clock cycle. Only relevant when memory configuration type is SDR2:1 or DDR. 0=No Extension 1=Extend

MEM_DQM_RD_DIS	26	0x0	Disable assertion of DQM for read commands. 0=Enabled 1=Disabled
MEM_REFRESH_DIS	27	0x1	Disable refresh cycles. 0=Enabled 1=Disabled
MEM_REFRESH_RATE	31:28	0x0	Refresh cycle rate set depending on XCLK frequency. 0= 10 MHz - 50 MHz (1 refresh every 156 XCLK's) 1= 50 MHz - 66 MHz (1 refresh every 781 XCLK's) 2= 66 MHz - 75 MHz (1 refresh every 1031 XCLK's) 3= 75 MHz - 83 MHz (1 refresh every 1172 XCLK's) 4= 83 MHz - 90 MHz (1 refresh every 1297 XCLK's) 5= 90 MHz - 95 MHz (1 refresh every 1406 XCLK's) 6= 95 MHz - 100 MHz (1 refresh every 1484 XCLK's) 7=100 MHz - 105 MHz (1 refresh every 1563 XCLK's) 8=105 MHz - 110 MHz (1 refresh every 1641 XCLK's) 9=110 MHz - 115 MHz (1 refresh every 1719 XCLK's) 10=115 MHz - 120 MHz (1 refresh every 1797 XCLK's) 11=120 MHz - 125 MHz (1 refresh every 1875 XCLK's) 12=125 MHz and above (1 refresh every 1953 XCLK's)

MEM_CNTL: Memory Control Register.

EXT_MEM_CNTL MMR:0x0144 MMR_1:0x0144 IND:0x0144 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_TRP	1:0	0x3	RAS Precharge time, or PRE to ACTV delay: 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
MEM_TRCD	3:2	0x3	RAS to CAS delay, or ACTV to command delay: 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks

MEM_TRAS	6:4	0x7	RAS low minimum pulse width, or ACTV to PRE delay of the 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
(reserved)	7		
MEM_TRRD	9:8	0x2	RAS to RAS delay, or ACTV to ACTV delay: 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
MEM_TR2W	11:10	0x1	Read to write data turnaround clock cycles: 0=0 clocks 1=1 clock 2=2 clocks 3=3 clocks
MEM_TWR	13:12	0x1	Write recovery time: 0=1 clock 1=2 clocks 2=3 clocks 3=4 clocks
MEM_TBWC	14	0x1	Block write cycle time: 0=1 clock 1=2 clocks
MEM_TSML	15	0x1	Special mode register write latency: 0=1 clock 1=2 clocks

MEM_TR2R	17:16	0x0	Read to read data turnaround time of 2 different memory parts driving the same MD signals: 0=0 clocks 1=1 clock 2=2 clocks 3=3 clocks
(reserved)	27:18		
MEM_TW2R_MODE	28	0x0	Write to Read command delay: 0=1 clock 1=use MEM_TWR for write to read command delay
MEM_TEST_MODE	30:29	0x0	Test mode for memory controller. No test mode was actually implemented. 0=Normal
(reserved)	31		

EXT_MEM_CNTL: Extended Memory Control Register.

MEM_INTF_CNTL MMR:0x014C MMR_1:0x014C IND:0x014C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_SSTL_EN	0	0x0	LVTTL/SSTL interface. 0=LVTTL interface 1=SSTL interface
MEM_MA_YCLK	1	0x0	Propagate memory address signals off of the falling edge of YCLK. 0=propagate off of XCLK 1=propagate off of YCLKb
MEM_CNTL_YCLK	2	0x0	Propagate RAS/CAS/WE/DSF signals off of the falling edge of YCLK. 0=propagate off of XCLK 1=propagate off of YCLKb

MEM_CS_YCLK	3	0x0	Propagate CS signals off of the falling edge of YCLK. 0=propagate off of XCLK 1=propagate off of YCLKb
MEM_HCLK0_DRIVE	4	0x0	Drive strength of HCLK0 pin. 0=low drive strength 1=high drive strength
MEM_HCLK1_DRIVE	5	0x0	Drive strength of HCLK1 pin. 0=low drive strength 1=high drive strength
MEM_MA_DRIVE	6	0x0	Drive strength of memory address pins. 0=low drive strength 1=high drive strength
MEM_CNTL_DRIVE	7	0x0	Drive strength of RAS/CAS/WE/DSF pins. 0=low drive strength 1=high drive strength
MEM_CS_DRIVE	8	0x0	Drive strength of CS pins. 0=low drive strength 1=high drive strength
MEM_QS_DRIVE	9	0x0	Drive strengths of QS pins. 0=low drive strength 1=high drive strength
MEM_DQML_DRIVE	10	0x0	Drive strength of DQM(7:0) pins. 0=low drive strength 1=high drive strength
MEM_DQMU_DRIVE	11	0x0	Drive strength of DQM(15:8) pins. 0=low drive strength 1=high drive strength

MEM_MDLE_DRIVE	12	0x0	Drive strength of even pins of MD(63:0). 0=low drive strength 1=high drive strength
MEM_MDLO_DRIVE	13	0x0	Drive strength of odd pins of MD(63:0). 0=low drive strength 1=high drive strength
MEM_MDUE_DRIVE	14	0x0	Drive strength of even pins of MD(127:64). 0=low drive strength 1=high drive strength
MEM_MDUO_DRIVE	15	0x0	Drive strength of odd pins of MD(127:64). 0=low drive strength 1=high drive strength
MEM_QS_REC	16	0x0	Receiver mode of QS pins. 0=hysteresis receiver 1=differential receiver
MEM_MD_REC	17	0x0	Receiver mode of MD pins. 0=hysteresis receiver 1=differential receiver
MEM_MA_SR	18	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_CNTL_SR	19	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_CS_SR	20	0x0	<No Description> 0=slow slew rate 1=fast slew rate

MEM_QS_SR	21	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_DQML_SR	22	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_DQMU_SR	23	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_MDLE_SR	24	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_MDLO_SR	25	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_MDUE_SR	26	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_MDUO_SR	27	0x0	<No Description> 0=slow slew rate 1=fast slew rate
MEM_MDLE_YCLK	28	0x0	<No Description> 0=propagate off of XCLK 1=propagate off of YCLKb
MEM_MDLO_YCLK	29	0x0	<No Description> 0=propagate off of XCLK 1=propagate off of YCLKb

MEM_DQML_YCLK	30	0x0	<No Description> 0=propagate off of XCLK 1=propagate off of YCLKb
MEM_TRISTATE_IO	31	0x0	<No Description> 0=normal 1=tristate

MEM_INTF_CNTL: Memory Interface Control Signals.

MEM_STR_CNTL MMR:0x0150 MMR_1:0x0150 IND:0x0150 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
STR0_SEL	2:0	0x0	Strobe signal for MD(31:0) 0=positive edge of XCLK 1=negative edge of XCLK 2=HCLK0 feedback 3=HCLK1 feedback 4=HCLK0 feedback by 2 5=HCLK1 feedback by 2 6=QS0 delayed 7=QS0 direct from pad
(reserved)	3		
STR1_SEL	6:4	0x0	Strobe signal for MD(63:32). 0=positive edge of XCLK 1=negative edge of XCLK 2=HCLK0 feedback 3=HCLK1 feedback 4=HCLK0 feedback by 2 5=HCLK1 feedback by 2 6=QS1 delayed 7=QS1 direct from pad
(reserved)	7		
STR2_SEL	10:8	0x0	Strobe signal for MD(95:64). 0=positive edge of XCLK 1=negative edge of XCLK 2=HCLK0 feedback 3=HCLK1 feedback
(reserved)	11		
STR3_SEL	14:12	0x0	Strobe signal for MD(127:96). 0=positive edge of XCLK 1=negative edge of XCLK 2=HCLK0 feedback 3=HCLK1 feedback
(reserved)	15		

HCLK0_FB_SKEW	18:16	0x0	Programmable delay of feedback signal selected by HCLK0_FB_SEL. Only has effect when HCLK0 feedback or HCLK0 feedback by 2 is selected as a read data strobe.
HCLK0_FB_SEL	19	0x0	Pin to use as HCLK0 feedback signal. 0=HCLK0 pin 1=QS0 pin
HCLK1_FB_SKEW	22:20	0x0	Programmable delay of feedback signal selected by HCLK1_FB_SEL. Only has effect when HCLK1 feedback or HCLK1 feedback by 2 is selected as a read data strobe.
HCLK1_FB_SEL	23	0x0	Pin to use as HCLK1 feedback signal. 0=HCLK1 pin 1=QS1 pin
MEM_QS_CONFIG	24	0x0	<No Description> 0=QS per 32 bits (2 strobes) 1=QS per 16 bits (4 strobes)
(reserved)	27:25		
MEM_DYN_HCLK_FBL	28	0x0	<No Description> 0=read strobe always enabled 1=read strobe dynamic, only enabled for reads
MEM_DYN_HCLK_FBU	29	0x0	<No Description> 0=read strobe always enabled 1=read strobe dynamic, only enabled for reads
MEM_DYN_STR	30	0x0	<No Description> 0=read strobe always enabled 1=read strobe dynamic, only enabled for reads
MEM_DYN_DREN	31	0x0	<No Description> 0=differential receiver always enabled if chosen 1=differential receiver dynamic, only enabled for reads

MEM_STR_CNTL: Memory Read Data Strobe Control.

MEM_INIT_LAT_TIMER MMR:0x0154 MMR_1:0x0154 IND:0x0154 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_PC0R_INIT_LAT	5:0	0x3f	Initial Latency for PC0R request.
MEM_PC0W_INIT_LAT	11:6	0x3f	Initial Latency for PC0W request.
MEM_PC1R_INIT_LAT	17:12	0x3f	Initial Latency for PC1R request.
MEM_PC1W_INIT_LAT	23:18	0x3f	Initial Latency for PC1W request.
MEM_TEXEL_INIT_LAT	29:24	0x3f	Initial Latency for Texel request.
(reserved)	31:30		

MEM_INIT_LAT_TIMER: Initial latency timer for memory controller arbiter.

3.20 DAC

<No description>

DAC_CNTL MMR:0x0058 MMR_1:0x0058 IOR:0x0058 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DAC_RANGE_CNTL	1:0	0x2	DAC control bits. Should be set to '10' by default. BIOS will modify if needed. 0=PAL Output Level 1=NTSC Output Level 2=PS2 Output Level 3=Reserved
DAC_BLANKING	2	0x0	Controls use of DAC blanking pedestal during horizontal and vertical blanks. 0 = 0 IRE blanking pedestal . 1 = Enable 7.5 IRE blanking pedestal. Increases display brightness relative to blanking regions.
DAC_CMP_EN	3	0x1	<No Description>
(reserved)	6:4		
DAC_CMP_OUTPUT (R)	7	0x0	DAC comparator output. 0 = At least 1 comparator > ~0.373V. 1 = All 3 comparators < ~0.373V. The compartors are used for monitor detection by sensing if the termination on the R,G&B lines is 75 ohms (no monitor) or 37.5 ohms (monitor present). This can also determine if the attached monitor is monchrome or color. To use this register the driver must ensure the raster is currently in the active display area. Reading multiple times is recommended. To test if Green is termintated, set Red and Blue to 0 and set Green to 0x5A (post palette). If the Green line is terminated, then DAC_CMP_OUTPUT will read back '1' when the raster is on the above color. See the programmers manual for more details on the monitor detection algorithm.

DAC_8BIT_EN	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 = 6 bit 1 = 8 bit
DAC_4BPP_PIX_ORDER	9	0x0	Selects the order of pixel nibbles within bytes for 4 bpp extended (non-VGA) display modes. 0 = Most significant nibble is the left pixel. 1 = Least significant nibble is the left pixel.
DAC_TVO_EN	10	0x0	Enables generation of TV output byte stream for use by ImpacTV/Ripper encoder. Use the MPP_TB_TVO_EN or MPP_GP_TVO_EN bits to control which MPP port drives out the display data. This depends on the board design. The display clock generation must also be programmed for TV out to get an image on the TV. 0=Disable 1=Enable
DAC_TVO_OVR_EXCL	11	0x0	Used when TV out active to suppress overscan on the CRT monitor. Overscan is used by the TV out circuitry for frame synchronization. 0=CRT & TVO overscan 1=TVO overscan only
DAC_TVO_16BPP_DITH_EN	12	0x0	Selects method of encoding TV out data when using 565 mode. Dither method is one dimensional error diffusion. 0=Disable 1=Enable Dithering on 16BPP TV Output
DAC_VGA_ADR_EN	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).
(reserved)	14		

DAC_PDWN	15	0x0	Power down internal DAC (DAC macro only). This 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. This should save about 56 mA.
(reserved)	18:16		
DAC_CRC_EN	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. 0 = Disable. Reset before using. 1 = Enable. CRC will start in next vertical blank, and run for one field/frame. Note: There is no hardware control of whether the CRC occurs on even or odd frames in interlaced modes. This can be done by software polling the CRTC_FRAME and CRTC_CRNT_VLINE registers before enabling the CRC. The CRC's for even and odd frames will be different.
(reserved)	23:20		
DAC_MASK	31:24	0xff	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.

DAC_CNTL: General control for the RGB DAC and palette.

DAC_CRC_SIG MMR:0x02CC MMR_1:0x02CC IND:0x02CC [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DAC_CRC_SIG	23:0	0x0	DAC CRC signature value. Use DAC_CRC_EN to initate a field or frame analysis. After compeletion of the field/frame the DAC_CRC_SIG will remain constant until DAC_CRC_EN is cleared and set again. Only the even or odd field of interlaced displays is CRC'ed at one time. This is the code for the CRC signature: CRCB(7:0) <= 0; CRCG(7:0) <= 0; CRCR(7:0) <= 0; While in frame to capture and not blank do once per pixel: CRCB(7:1) <= Blue(7:1) xor CRCB(6:0); CRCB(0) <= (Blue(0) xor CRCB(0)) xor (CRCB(7) xor CRCG(7)); CRCG(7:1) <= Green(7:1) xor CRCG(6:0); CRCG(0) <= (Green(0) xor CRCG(0)) xor (CRCG(7) xor CRCR(7)); CRCR(7:1) <= Red(7:1) xor CRCR(6:0); CRCR(0) <= Red(0) xor (CRCR(0) xor CRCR(7)); End do; DAC_CRC_SIG(23:0) <= CRCB(7:0) & CRCG(7:0) & CRCR(7:0);
(reserved)	31:24		

DAC_CRC_SIG: CRC signature value

3.21 External DAC Support

<No description>

3.22 Full Custom Macros

<No description>

3.23 Test and Debug

<No description>

TEST_DEBUG_CNTL MMR:0x0120 MMR_1:0x0120 IND:0x0120 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
TEST_DEBUG_OUT_EN	0	0x0	Enables test & debug output bus on AMC connector pins. 0=Disable 1=Enable
TEST_DEBUG_IN_LOW_EN	1	0x0	Enables lower test & debug input bus on video port pins. 0=Disable 1=Enable
TEST_DEBUG_IN_HIGH_EN	2	0x0	Enables upper test & debug input bus on video port pins. 0=Disable 1=Enable
TEST_IDDQ_EN	3	0x0	Sets the device into quiescent current mode. Disables built-in pull-up and pull-down resistors. 0=Disable 1=Enable
TEST_VIDBLK	4	0x0	<No Description> 0=Disable 1=Enable
TEST_PM4	5	0x0	<No Description> 0=Disable 1=Enable
TEST_DELAY_RING	6	0x0	<No Description> 0=Disable 1=Enable

TEST_MREG	7	0x0	Enables merged register bus controller to assert data on the test & dubug output bus. 0=Disable 1=Enable
TEST_PLL	8	0x0	<No Description> 0=Disable 1=Enable
TEST_DISPENG	9	0x0	<No Description> 0=Disable 1=Enable
TEST_RAMDAC	10	0x0	<No Description> 0=Disable 1=Enable
TEST_MEMCTLR	11	0x0	Select Memory Controller debug signals. 0=Disable 1=Enable
TEST_HBIU	12	0x0	<No Description> 0=Disable 1=Enable
TEST_AGP	13	0x0	<No Description> 0=Disable 1=Enable
TEST_2D_GUI	14	0x0	<No Description> 0=Disable 1=Enable
TEST_3D_GUI	15	0x0	<No Description> 0=Disable 1=Enable

(reserved)	16		
TEST_HOSTPATH	17	0x0	Enables host path controller to assert data on the test & debug output bus. 0=Disable 1=Enable
TEST_CMDFIFO_LOCK	18	0x0	<No Description> 0=Disable 1=Enable
TEST_BLOCK_GUI_INITIATORS	19	0x0	<No Description> 0=Disable 1=Enable
TEST_BLOCK_PM4_INITIATORS	20	0x0	<No Description> 0=Disable 1=Enable
TEST_VIPH	21	0x0	<No Description> 0=Disable 1=Enable
TEST_MPPTB	22	0x0	<No Description> 0=Disable 1=Enable
TEST_MPPGP	23	0x0	<No Description> 0=Disable 1=Enable
TEST_I2C	24	0x0	<No Description> 0=Disable 1=Enable
(reserved)	25		

TEST_SUBPIC	26	0x0	<No Description> 0=Disable 1=Enable
TEST_OV0SCALE	27	0x0	<No Description> 0=Disable 1=Enable
TEST_LCDENG	28	0x0	When enabled and TEST_DEBUG_MUX=0, the DAC_CRC_SIG will be based on the display data after the LCD frame modulation. 0=Disable 1=Enable
TEST_PC	29	0x0	<No Description> 0=Disable 1=Enable
TEST_TMDS	30	0x0	<No Description> 0=Disable 1=Enable
(reserved)	31		

TEST_DEBUG_CNTL: Test and debug mode control.

TEST_DEBUG_MUX MMR:0x0124 MMR_1:0x0124 IND:0x0124 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
TEST_DEBUG_MUX	3:0	0x0	Debug mode selection bits. Function dependent on which test mode is activated in TEST_DEBUG_CNTL.
TEST_DEBUG_BANK	5:4	0x0	Additional debug mode selection bits. Same purpose as TEST_DEBUG_MUX.
(reserved)	7:6		

TEST_DEBUG_CLK	12:8	0x0	Selects the clock signal to mux out as follows: 0 = no clock, output 0. 1 = Oscillator macro internal output (Xtalin) 2 = PPllClk/2 3 = PPll reference divider output 4 = PPll feedback divider output 5 = PPllClk output (slipable) 6 = PPllClk output (fixed) 9 = PCLK (Dispeng word clock) A = ECP (Overlay/scalar clock) B = XPllClk output C = XPllClk/2 D = XPll & MPll reference divider output E = XPll feedback divider output F = XCLK (memory controller main internal clock) 10 = YCLK (memory controller 2x clock for DDR and fast SDR) 11 = DLL test clock 0 12 = DLL test clock 1 13 = MPllClk output 14 = MPllClk/2 15 = MPll feedback divider output 16 = MCLK (non-dynamic version of main engine clock) 17 = GCP (dynamic 2D engine clock) 18 = RCP (dynamic register read/write clock) 19 = PIPE3D_CP (dynamic 3D engine clock) 1A = X1CLK (AGP interface X1 clock) 1B = X2CLK (AGP interface X2 clock) 1C = BCLK (main host interface clock) 1D = F1CP (video capture port 1 clock) 1E = F0CP (video capture port 0 clock) 1F = PLFBCLK (panel line feedback clock)
(reserved)	14:13		
TEST_DEBUG_CLK_INV	15	0x0	Enables inversion of test clock output. 0=Non-inverted 1=Inverted
(reserved)	31:16		

TEST_DEBUG_MUX: Mux & clock controls for test & debug modes.

HW_DEBUG MMR:0x0128 MMR_1:0x0128 IND:0x0128 **[RW] 32 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**

HW_0_DEBUG	0	0x0	<No Description> 0=Not allow register write requests to overrun memory write requests to hostpath 1=allow it
HW_1_DEBUG	1	0x0	<No Description> 0=HBLANK_END in VGA Horizontal Auto Ratio is calculated like in Mobility chip 1=HBLANK_END in VGA Horizontal Auto Ratio behaves as pre-ECO
HW_2_DEBUG	2	0x0	<No Description> 0=Normal 1=Enable forced RDY on retried slave writes
HW_3_DEBUG	3	0x0	<No Description> 0=Normal 1=Enable FILL STALL FLUSH FIX in the Pixel Cache
HW_4_DEBUG	4	0x0	<No Description>
HW_5_DEBUG	5	0x0	<No Description>
HW_6_DEBUG	6	0x0	<No Description> 0=Normal 1=force a soft reset for pm
HW_7_DEBUG	7	0x0	<No Description> 0=Auxiliary Window output is active low 1=Auxiliary Window output is active high
HW_8_DEBUG	8	0x0	<No Description> 0=Normal 1=force a soft reset for eng_pix

HW_9_DEBUG	9	0x0	<No Description> 0=Test input data for TMDS macro is from TMDS_TSTPIX register 1=Test input data for TMDS macro is from Zero-One pattern generator
HW_A_DEBUG	10	0x0	<No Description> 0=Normal 1=Disable CULL function in ipu (pm4)
HW_B_DEBUG	11	0x0	<No Description>
HW_C_DEBUG	12	0x0	<No Description>
HW_D_DEBUG	13	0x0	<No Description> 0=Normal 1=Force AGP accesses to tiled surfaces to be interpreted vertically, but fetched horizontally.
HW_E_DEBUG	14	0x0	<No Description> 0=Normal 1=Force texture L0 cache into optimization mode, instead of using TOP/BOT, use N/N+1.
HW_F_DEBUG	15	0x0	<No Description>
HW_10_DEBUG	16	0x0	<No Description>
HW_11_DEBUG	17	0x0	<No Description> 0=Normal 1=Active HIGH disable FB_AGP_XCHECK
HW_12_DEBUG	18	0x0	<No Description>
HW_13_DEBUG	19	0x0	<No Description>
HW_14_DEBUG	20	0x0	<No Description>
HW_15_DEBUG	21	0x0	<No Description>

HW_16_DEBUG	22	0x0	<No Description>
HW_17_DEBUG	23	0x0	<No Description>
HW_18_DEBUG	24	0x0	<No Description>
HW_19_DEBUG	25	0x0	<No Description>
HW_1A_DEBUG	26	0x0	<No Description>
HW_1B_DEBUG	27	0x0	<No Description>
HW_1C_DEBUG	28	0x0	<No Description>
HW_1D_DEBUG	29	0x0	<No Description>
HW_1E_DEBUG	30	0x0	<No Description>
HW_1F_DEBUG	31	0x0	<No Description>

HW_DEBUG: For use in chip debugging and minor revisions.

3.24 Destination GUI Registers

<No description>

DST_OFFSET MMR:0x1404 MMR_1:0x1404 IND:0x1404 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_OFFSET	25:0	0x0	Byte-aligned destination offset address. This is a virtual address. The lower 32MB maps to frame buffer, the upper 32MB to AGP_BASE + DST_OFFSET(24:0). Note that this register is constrained to 128 bit alignment. NOTE: Bits 3:0 of this field are hardwired to ZERO
(reserved)	31:26		

DST_OFFSET: <No Description>

DST_PITCH_OFFSET MMR:0x142C MMR_1:0x142C IND:0x142C [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_OFFSET	20:0	0x0	32 byte-aligned destination offset address.
DST_PITCH	30:21	0x0	Destination pitch in multiple of 8 pixels.
DST_TILE	31	0x0	Destination tile bit.

DST_PITCH_OFFSET: <No Description>

DST_PITCH MMR:0x1408 MMR_1:0x1408 IND:0x1408 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_PITCH	9:0	0x0	Destination pitch in pixels*8. Note that in 8 bpp modes, the pitch must be a multiple of 16 pixels. Note that this field in programmed in bytes*8 for 24 bpp modes.
(reserved)	15:10		
DST_TILE	16	0x0	Denotes whether the destination surface is in 'tiled' format.

DST_PITCH_ADJ	18:17	0x0	Denotes that DST_PITCH should be multiplied prior to use:
(reserved)	31:19		

DST_PITCH: <No Description>

DST_X MMR:0x141C MMR_1:0x141C IND:0x141C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_X	13:0	0x0	Destination X co-ordinate (range -8192 to 8191) If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.12.0 bit integer with bits 14:13 reserved.
(reserved)	31:14		

DST_X: <No Description>

DST_Y MMR:0x1420 MMR_1:0x1420 IND:0x1420 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_Y	13:0	0x0	Destination Y coordinate (range -8192 to 8191) - Bits 15:14 should be copies of bit 13 (i.e. sign extended). If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.14.0 integer.
(reserved)	31:14		

DST_Y: <No Description>

DST_X_Y MMR:0x1594 MMR_1:0x1594 IND:0x1594 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_Y	13:0	0x0	Destination Y coordinate (range -8192 to 8191) - Bits 15:14 should be copies of bit 13 (i.e. sign extended). If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.14.0 integer.
(reserved)	15:14		

DST_X	29:16	0x0	Destination X co-ordinate (range -8192 to 8192) If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.12.0 bit integer with bits 14:13 reserved.
(reserved)	31:30		

DST_X_Y: <No Description>

DST_Y_X MMR:0x1438 MMR_1:0x1438 IND:0x1438 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_X	13:0	0x0	If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.12.0 bit integer with bits 14:13 reserved.
(reserved)	15:14		
DST_Y	29:16	0x0	If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.14.0 integer.
(reserved)	31:30		

DST_Y_X: <No Description>

DST_WIDTH MMR:0x140C MMR_1:0x140C IND:0x140C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_WIDTH	13:0	0x0	Destination width. Only bits 12:0 are used for rectangle draws. Bit 15 is write ONLY and will always read back as '0'. Bits [15:13] are aliased to DST_BRES_LENGTH[15:13] and are used for trapezoid draw operations.
(reserved)	31:14		

DST_WIDTH: <No Description>

DST_HEIGHT MMR:0x1410 MMR_1:0x1410 IND:0x1410 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_HEIGHT	13:0	0x0	Destination height

(reserved)	31:14		

DST_HEIGHT: <No Description>

DST_HEIGHT_WIDTH MMR:0x143C MMR_1:0x143C IND:0x143C [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_WIDTH	13:0	0x0	Destination width
(reserved)	15:14		
DST_HEIGHT	29:16	0x0	<No Description>
(reserved)	31:30		

DST_HEIGHT_WIDTH: <No Description>

DST_WIDTH_HEIGHT MMR:0x1598 MMR_1:0x1598 IND:0x1598 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_HEIGHT	13:0	0x0	Destination height (bits 12:0 aliased to TRAIL_X@DST_BRES_LNTH)
(reserved)	15:14		
DST_WIDTH	29:16	0x0	Destination width. Only bits 12:0 are used for rectangle draws. Bit 15 is write ONLY and will always read back as '0'. Bits [15:13] are aliased to DST_BRES_LENGTH[15:13] and are used for trapezoid draw operations.
(reserved)	31:30		

DST_WIDTH_HEIGHT: <No Description>

DST_HEIGHT_WIDTH_8 MMR:0x158C MMR_1:0x158C IND:0x158C [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	15:0		
DST_WIDTH	23:16	0x0	Destination width:range 0 to 256 (ZERO extent)
DST_HEIGHT	31:24	0x0	Destination width:range 0 to 256 (ZERO extent)

DST_HEIGHT_WIDTH_8: <no active register>

DST_HEIGHT_Y MMR:0x15A0 MMR_1:0x15A0 IND:0x15A0 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_Y	13:0	0x0	Destination Y
(reserved)	15:14		
DST_HEIGHT	29:16	0x0	Destination Height
(reserved)	31:30		

DST_HEIGHT_Y: <No Description>

DST_WIDTH_X MMR:0x1588 MMR_1:0x1588 IND:0x1588 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_X	13:0	0x0	Destination X coordinate. If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.12.0 bit integer with bits 14:13 reserved.
(reserved)	15:14		
DST_WIDTH	29:16	0x0	Destination width. Only bits 12:0 are used for rectangle draws. Bit 15 is write ONLY and will always read back as '0'. Bits [15:13] are aliased to DST_BRES_LENGTH[15:13] and are used for trapezoid draw operations.
(reserved)	31:30		

DST_WIDTH_X: <No Description>

DST_WIDTH_X_INCY MMR:0x159C MMR_1:0x159C IND:0x159C [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_X	13:0	0x0	Destination X coordinate. If SUB_PIX_ON is set, this field is interpreted as a S.12.2 number. Otherwise it is a S.12.0 bit integer with bits 14:13 reserved.
(reserved)	15:14		

DST_WIDTH	29:16	0x0	Destination width. Only bits 12:0 are used for rectangle draws. Bit 15 is write ONLY and will always read back as '0'. Bits [15:13] are aliased to DST_BRES_LENGTH[15:13] and are used for trapezoid draw operations.
(reserved)	31:30		

DST_WIDTH_X_INCY: <No Description>

DST_BRES_LNTH MMR:0x1634 MMR_1:0x1634 IND:0x1634 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_BRES_LNTH	13:0	0x0	Bresenham line length.
(reserved)	31:14		

DST_BRES_LNTH: <No Description>

DST_BRES_ERR MMR:0x1628 MMR_1:0x1628 IND:0x1628 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_BRES_ERR	19:0	0x0	Bresenham error term for line and Trapezoid leading edge
(reserved)	31:20		

DST_BRES_ERR: <No Description>

DST_BRES_INC MMR:0x162C MMR_1:0x162C IND:0x162C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_BRES_INC	19:0	0x0	Bresenham increment for line and Trapezoid leading edge
(reserved)	31:20		

DST_BRES_INC: <No Description>

DST_BRES_DEC MMR:0x1630 MMR_1:0x1630 IND:0x1630 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

DST_BRES_DEC	19:0	0x0	Bresenham decrement for line and Trapezoid leading edge
(reserved)	31:20		

DST_BRES_DEC: <No Description>

DST_X_SUB MMR:0x15A4 MMR_1:0x15A4 IND:0x15A4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
LEAD_X_FRACT	3:0	0x0	Sub pixel bits of Destination X coordinate. Note that when DST_X is written these bits are set to 1000 (one half)
LEAD_X	17:4	0x0	Destination X coordinate: range -8192 to 8191. Aliased to DST_X[13:0]
(reserved)	31:18		

DST_X_SUB: <No Description>

DST_Y_SUB MMR:0x15A8 MMR_1:0x15A8 IND:0x15A8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
LEAD_Y_FRACT	3:0	0x0	Sub pixel bits of Destination Y coordinate Note that when DST_Y is written these bits are set to 1000 (one half)
LEAD_Y	17:4	0x0	Destination Y coordinate: range -8192 to 8191. Aliased to DST_Y[13:0]
(reserved)	31:18		

DST_Y_SUB: <No Description>

DST_BRES_LNTH_SUB MMR:0x1638 MMR_1:0x1638 IND:0x1638 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_BRES_LNTH_SUB	3:0	0x0	Bresenham line length.
DST_BRES_LNTH	17:4	0x0	Bresenham line, and Trapezoid leading edge length. This field is aliased with DST_BRES_LNTH[13:0].
(reserved)	31:18		

DST_BRES_LNTH_SUB: <No Description>

COMPOSITE_SHADOW_ID MMR:0x1A0C MMR_1:0x1A0C IND:0x1A0C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
COMPOSITE_SHADOW_ID	23:0	0x0	This field is a count of 3D primatives executed. It is used as part of the shadow ID algorithm, but may also be used as a general counter for performance purposes.
(reserved)	26:24		
COMPOSITE_SHADOW_AUTO_INC_DIS	27	0x0	<No Description> 0=Increment the COMPOSITE_SHADOW_ID field after each primitive is executed. 1=COMPOSITE_SHADOW_ID field does not increment.
(reserved)	31:28		

COMPOSITE_SHADOW_ID: Triangle count for shadow algorithm.

DST_PITCH_OFFSET_C MMR:0x1C80 MMR_1:0x1C80 IND:0x1C80 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_OFFSET	20:0	0x0	32 byte-aligned destination offset address
DST_PITCH	30:21	0x0	Destination pitch in pixels *8. Note that for monochrome modes the destination pitch must be a multiple of 64 pixels.
DST_TILE	31	0x0	Destination tile bit

DST_PITCH_OFFSET_C: Aliased to DST_ PITCH_OFFSET.

3.25 Source GUI Registers

<No description>

SRC_OFFSET MMR:0x15AC MMR_1:0x15AC IND:0x15AC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_OFFSET	25:0	0x0	Byte aligned source offset address. This is a 64M virtual address (see DST_OFFSET definition for more details) . Note that this register is constrained to be 128 bit aligned. Note that bits 3:0 are hardwired to 0. NOTE: Bits 3:0 of this field are hardwired to ZERO
(reserved)	31:26		

SRC_OFFSET: <No Description>

SRC_PITCH_OFFSET MMR:0x1428 MMR_1:0x1428 IND:0x1428 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_OFFSET	20:0	0x0	32 byte-aligned source offset address. (i.e. 25:5 & '00000') byte address
SRC_PITCH	30:21	0x0	Source pitch
SRC_TILE	31	0x0	Source tile bit.

SRC_PITCH_OFFSET: <No Description>

SRC_PITCH MMR:0x15B0 MMR_1:0x15B0 IND:0x15B0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_PITCH	9:0	0x0	Source pitch in pixelsx8. Note that in monochrome mode the source pitch must be a multiple of 128 pixels. In 8bpp mode, source pitch must be a multiple of 16 pixels.
(reserved)	15:10		
SRC_TILE	16	0x0	Denotes whether the SRC memory is in 'tiled' format

(reserved)	31:17		

SRC_PITCH: <No Description>

SRC_X MMR:0x1414 MMR_1:0x1414 IND:0x1414 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_X	13:0	0x0	Source X coordinate: range -8192 to 8191
(reserved)	31:14		

SRC_X: <No Description>

SRC_Y MMR:0x1418 MMR_1:0x1418 IND:0x1418 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_Y	13:0	0x0	Source Y coordinate: range -8192 to 8191
(reserved)	31:14		

SRC_Y: <No Description>

SRC_X_Y MMR:0x1590 MMR_1:0x1590 IND:0x1590 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_Y	13:0	0x0	Source Y coordinate: range -8192 to 8191
(reserved)	15:14		
SRC_X	29:16	0x0	Source X coordinate: range -8192 to 8191
(reserved)	31:30		

SRC_X_Y: <No Description>

SRC_Y_X MMR:0x1434 MMR_1:0x1434 IND:0x1434 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_X	13:0	0x0	Source X coordinate: range -8192 to 8191

(reserved)	15:14		
SRC_Y	29:16	0x0	Source Y coordinate: range -8192 to 8191
(reserved)	31:30		

SRC_Y_X: <No Description>

SRC_SC_RIGHT MMR:0x1654 MMR_1:0x1654 IND:0x1654 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_SC_RIGHT	13:0	0x0	Right scissor: range -8192 to 8191
(reserved)	31:14		

SRC_SC_RIGHT: <No Description>

SRC_SC_BOTTOM MMR:0x165C MMR_1:0x165C IND:0x165C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_SC_BOTTOM	13:0	0x0	Bottom scissor: range -8192 to 8191
(reserved)	31:14		

SRC_SC_BOTTOM: <No Description>

SRC_SC_BOTTOM_RIGHT MMR:0x16F4 MMR_1:0x16F4 IND:0x16F4 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SRC_SC_RIGHT	13:0	0x0	<No Description>
(reserved)	15:14		
SRC_SC_BOTTOM	29:16	0x0	<No Description>
(reserved)	31:30		

SRC_SC_BOTTOM_RIGHT: See SRC_SC_BOTTOM, SRC_SC_RIGHT

3.26 Host Data Registers

The host data registers provide pixel data which is utilized in the current drawing operation. The pixel data may be used as a monochrome pixel source or color pixel source. For rectangular drawing operations the pixel data may be either packed from one horizontal line to the next or unpacked. Sixteen 32 bit host data register are provided. All registers are treated identically and data is fed to the engine in the order in which it is written to any of the host data registers. Up to sixteen host data registers are provided to allow block data moves of variable length up to the depth of the parameter FIFO.

HOST_PATH_CNTL MMR:0x0130 MMR_1:0x0130 IND:0x0130 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MREG_ADDR_DELAY	1:0	0x3	Sets the time that the address and related signals will be valid before the read or write strobe is asserted for on-chip registers. This is set by the BIOS and should not be changed. Too low a value will corrupt register writes and reads. This affects writes to all non-GUI (not FIFOed) registers, and reads to all registers. 0=1 MCLK address prop. time 1=2 MCLK address prop. time 2=3 MCLK address prop. time 3=4 MCLK address prop. time
MREG_RD_DELAY	3:2	0x3	Sets the time the register files have to respond to read requests before the data is expected to be asserted on the shared read bus. This value will be set by the BIOS and should not be changed. Setting it too low could corrupt the register reads. This affects all on-chip registers. 0=1 MCLK reg. file read time 1=2 MCLK reg. file read time 2=3 MCLK reg. file read time 3=4 MCLK reg. file read time

MREG_RD_RETURN	6:4	0x7	Sets the time for register read data to propagate up the read bus to the host interface. Will be set by the BIOS and should not be changed. Register reads could be corrupted if set too low. This affects all on-chip registers. 0=1 MCLK read bus prop. time 1=2 MCLK read bus prop. time 2=3 MCLK read bus prop. time 3=4 MCLK read bus prop. time 4=5 MCLK read bus prop. time 5=6 MCLK read bus prop. time 6=7 MCLK read bus prop. time 7=8 MCLK read bus prop. time
(reserved)	7		
MREG_ARB_CNTL	9:8	0x0	Determines how access is granted to the internal merged non-GUI register bus when both the host (PCI/AGP) bus and the ProMo4 parser are both trying to use it at the same time. This does not affect access to FIFOed GUI registers. 0=Round robin host and GUI 1=Host wins over GUI 2=GUI wins over host
(reserved)	11:10		
MREG_RCP_EXT	14:12	0x2	Sets number of extra MCLK cycles RCP will remain running after the merged register bus is idle. Larger values will consume more power, but will save a clock cycle to turn RCP back on for subsequent register cycles. The BIOS will set this register, and further change should not be needed. 0=0 clocks 1=8 clocks 2=16 clocks 3=24 clocks 4=32 clocks 5=40 clocks 6=48 clocks 7=56 clocks
(reserved)	23:15		

HP_LIN_RD_CACHE_DIS	24	0x0	Selects whether to try to service linear aperture slave reads using data from previous read. 0=Linear aperture slave reads taken from hostpath cache, if possible. 1=Linear aperture slave reads always sent to pixel cache.
HP_VGA_RD_CACHE_DIS	25	0x0	Selects whether to try to service VGA aperture reads using the data from the previous read. 0=VGA aperture slave reads taken from hostpath cache, if possible. 1=VGA aperture slave reads always sent to pixel cache.
(reserved)	30:26		
HP_TEST_RST_CNTL	31	0x0	For HW test and debugging only. No use to software.

HOST_PATH_CNTL: Controls for the Merged Register Bus internal to the controller for non-FIFOed register writes, and all register reads.

HOST_DATA0 MMR:0x17C0 MMR_1:0x17C0 IND:0x17C0 **[W] 32 bits (access: 32)**			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA0	31:0	0x0	Host data register

HOST_DATA0: Host data register

HOST_DATA1 MMR:0x17C4 MMR_1:0x17C4 IND:0x17C4 **[W] 32 bits (access: 32)**			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA1	31:0	0x0	Host data register

HOST_DATA1: Host data register

HOST_DATA2 MMR:0x17C8 MMR_1:0x17C8 IND:0x17C8 **[W] 32 bits (access: 32)**			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA2	31:0	0x0	Host data register

HOST_DATA2: Host data registe

HOST_DATA3 MMR:0x17CC MMR_1:0x17CC IND:0x17CC [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA3	31:0	0x0	Host data register

HOST_DATA3: Host data register

HOST_DATA4 MMR:0x17D0 MMR_1:0x17D0 IND:0x17D0 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA4	31:0	0x0	Host data register

HOST_DATA4: Host data register

HOST_DATA5 MMR:0x17D4 MMR_1:0x17D4 IND:0x17D4 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA5	31:0	0x0	Host data register

HOST_DATA5: Host data register

HOST_DATA6 MMR:0x17D8 MMR_1:0x17D8 IND:0x17D8 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA6	31:0	0x0	Host data register

HOST_DATA6: Host data register

HOST_DATA7 MMR:0x17DC MMR_1:0x17DC IND:0x17DC [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA7	31:0	0x0	Host data register

HOST_DATA7: Host data register

HOST_DATA_LAST MMR:0x17E0 MMR_1:0x17E0 IND:0x17E0 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
HOST_DATA_LAST	31:0	0x0	Indicates the last host data register write for operation.

HOST_DATA_LAST: Host data register

3.27 Pattern Registers

Two pattern registers support three fixed destination aligned pattern modes; monochrome 8x8, 8bpp color 4x2, and 8bpp color 8x1. For the VT/GT-B, 8x8x8 patterns or brushes can be using a linear source in conjunction with the SRC_8x8x8_BRUSH@SRC_CNTL. For all patterns, the alignment of register data to the least significant bits of DST_X and DST_Y is as follows:

3.28 Scissor Registers

The scissor registers define the rectangular region within which data is drawn. Left and right scissor registers are within the range -4096 to +4095. Top and bottom scissor registers are within the range -16384 to +16383. Polylines which follow a trajectory to the left of the left scissor register will result in a line drawn along the left scissor coordinate.

SC_LEFT MMR:0x1640 MMR_1:0x1640 IND:0x1640 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_LEFT	13:0	0x0	Destination left scissor: range -8192 to 8191
(reserved)	31:14		

SC_LEFT: <No Description>

SC_RIGHT MMR:0x1644 MMR_1:0x1644 IND:0x1644 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT	13:0	0x0	Destination right scissor: range -8192 to 8191
(reserved)	31:14		

SC_RIGHT: <No Description>

SC_TOP MMR:0x1648 MMR_1:0x1648 IND:0x1648 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_TOP	13:0	0x0	Destination top scissor: range -8192 to 8191
(reserved)	31:14		

SC_TOP: <No Description>

SC_BOTTOM MMR:0x164C MMR_1:0x164C IND:0x164C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_BOTTOM	13:0	0x0	Destination bottom scissor: range -8192 to 8191
(reserved)	31:14		

SC_BOTTOM: <No Description>

AUX_SC_CNTL MMR:0x1660 MMR_1:0x1660 IND:0x1660 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX1_SC_ENB	0	0x0	Enable for Auxilliary 1 scissors 0 = Off 1 = On This bit is set to 0 on Chip Reset. 0=Off 1=On
AUX1_SC_MODE	1	0x0	Auxilliary Scissors can function in 1 of 2 modes: 0 = Additive. Combine with other destination SCISSORs with 'OR' 1 =Subtractive. Combine with other destination SCISSORs with 'AND NOT' 0=Additive 1=Subtractive
AUX2_SC_ENB	2	0x0	Enable for Auxilliary 2 scissors 0 = Off 1 = On This bit is set to 0 on Chip Reset 0=Off 1=On
AUX2_SC_MODE	3	0x0	Auxilliary Scissors can function in 1 of 2 modes: 0 = Additive. Combine with other destination SCISSORs with 'OR' 1 =Subtractive. Combine with other destination SCISSORs with 'AND NOT' 0=Additive 1=Subtractive

AUX3_SC_ENB	4	0x0	Enable for Auxilliary 3 scissors 0 = Off 1 = On This bit is set to 0 on Chip Reset 0=Off 1=On
AUX3_SC_MODE	5	0x0	Auxilliary Scissors can function in 1 of 2 modes: 0 = Additive. Combine with other destination SCISSORs with 'OR' 1 =Subtractive. Combine with other destination SCISSORs with 'AND NOT' 0=Additive 1=Subtractive
(reserved)	31:6		

AUX_SC_CNTL: <No Description>

AUX1_SC_LEFT MMR:0x1664 MMR_1:0x1664 IND:0x1664 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX1_SC_LEFT	13:0	0x0	Auxilliary 1 left scissor: range -8192 to 8191
(reserved)	31:14		

AUX1_SC_LEFT: <No Description>

AUX1_SC_RIGHT MMR:0x1668 MMR_1:0x1668 IND:0x1668 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX1_SC_RIGHT	13:0	0x0	Auxilliary 1 right scissor: range -8192 to 8191
(reserved)	31:14		

AUX1_SC_RIGHT: <No Description>

AUX1_SC_TOP MMR:0x166C MMR_1:0x166C IND:0x166C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX1_SC_TOP	13:0	0x0	Auxilliary 1 top scissor: range -8192 to 8191
(reserved)	31:14		

AUX1_SC_TOP: <No Description>

AUX1_SC_BOTTOM MMR:0x1670 MMR_1:0x1670 IND:0x1670 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX1_SC_BOTTOM	13:0	0x0	Auxilliary 2 bottom scissor: range -8192 to 8191
(reserved)	31:14		

AUX1_SC_BOTTOM: <No Description>

AUX2_SC_LEFT MMR:0x1674 MMR_1:0x1674 IND:0x1674 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX2_SC_LEFT	13:0	0x0	Auxilliary 2 left scissor: range -8192 to 8191
(reserved)	31:14		

AUX2_SC_LEFT: <No Description>

AUX2_SC_RIGHT MMR:0x1678 MMR_1:0x1678 IND:0x1678 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX2_SC_RIGHT	13:0	0x0	Auxilliary 2 right scissor: range -8192 to 8191
(reserved)	31:14		

AUX2_SC_RIGHT: <No Description>

AUX2_SC_TOP MMR:0x167C MMR_1:0x167C IND:0x167C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

AUX2_SC_TOP	13:0	0x0	Auxilliary 2 top scissor: range -8192 to 8191
(reserved)	31:14		

AUX2_SC_TOP: <No Description>

AUX2_SC_BOTTOM MMR:0x1680 MMR_1:0x1680 IND:0x1680 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX2_SC_BOTTOM	13:0	0x0	Auxilliary 2 bottom scissor: range -8192 to 8191
(reserved)	31:14		

AUX2_SC_BOTTOM: <No Description>

AUX3_SC_LEFT MMR:0x1684 MMR_1:0x1684 IND:0x1684 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX3_SC_LEFT	13:0	0x0	Auxilliary 3 left scissor: range -8192 to 8191
(reserved)	31:14		

AUX3_SC_LEFT: <No Description>

AUX3_SC_RIGHT MMR:0x1688 MMR_1:0x1688 IND:0x1688 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX3_SC_RIGHT	13:0	0x0	Auxilliary 3 right scissor: range -8192 to 8191
(reserved)	31:14		

AUX3_SC_RIGHT: <No Description>

AUX3_SC_TOP MMR:0x168C MMR_1:0x168C IND:0x168C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX3_SC_TOP	13:0	0x0	Auxilliary 3 top scissor: range -8192 to 8191
(reserved)	31:14		

AUX3_SC_TOP: <No Description>

AUX3_SC_BOTTOM MMR:0x1690 MMR_1:0x1690 IND:0x1690 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
AUX3_SC_BOTTOM	13:0	0x0	Auxilliary 3 bottom scissor: range -8192 to 8191
(reserved)	31:14		

AUX3_SC_BOTTOM: <No Description>

SECONDARY_SCALE_PITCH MMR:0x1980 MMR_1:0x1980 IND:0x1980 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SECONDARY_SCALE_PITCH	8:0	0x0	Pitch in pixels*8 of the scalar source data for RGB and packed modes. The pitch is required to be programmed so that all source lines are an integer number of QWORDs
(reserved)	15:9		
SECONDARY_SCALE_TILE	16	0x0	
SECONDARY_SCALE_OFFSET_PTR	20:17	0x0	Denotes which of SECONDARY_TEX_OFFSET 0 - 7 to use as the SECONDARY_SCALE_OFFSET
(reserved)	29:21		
SECONDARY_SCALE_PITCH_ADJ	31:30	0x0	Denotes that SECONDARY_SCALE_PITCH should be multiplied prior to use. 0 = no adjustment on SECONDARY_SCALE_PITCH. 1 = multiply SECONDARY_SCALE_PITCH by 2 prior to use 2 = multiply SECONDARY_SCALE_PITCH by 4 prior to use. 3 = (Reserved).

SECONDARY_SCALE_PITCH: <No Description>

SECONDARY_SCALE_X_INC MMR:0x1984 MMR_1:0x1984 IND:0x1984 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
SECONDARY_SCALE_X_INC	19:4	0x0	<No Description>

(reserved)	31:20		

SECONDARY_SCALE_X_INC: <No Description>

SECONDARY_SCALE_HACC MMR:0x198C MMR_1:0x198C IND:0x198C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
SECONDARY_SCALE_HACC	27:4	0x0	<No Description>
(reserved)	31:28		

SECONDARY_SCALE_HACC: <No Description>

SECONDARY_SCALE_Y_INC MMR:0x1988 MMR_1:0x1988 IND:0x1988 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
SECONDARY_SCALE_Y_INC	19:4	0x0	Y accumulator increment, 12 bits fractional, 4 bits unsigned integer.
(reserved)	31:20		

SECONDARY_SCALE_Y_INC: <No Description>

SECONDARY_SCALE_VACC MMR:0x1990 MMR_1:0x1990 IND:0x1990 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
SECONDARY_SCALE_VACC	26:4	0x0	<No Description>
(reserved)	31:27		

SECONDARY_SCALE_VACC: <No Description>

SC_TOP_LEFT MMR:0x16EC MMR_1:0x16EC IND:0x16EC [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

SC_LEFT	13:0	0x0	
(reserved)	15:14		
SC_TOP	29:16	0x0	Top scissor
(reserved)	31:30		

SC_TOP_LEFT: Destination SC_TOP_LEFT

SC_BOTTOM_RIGHT MMR:0x16F0 MMR_1:0x16F0 IND:0x16F0 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT	13:0	0x0	Right scissor
(reserved)	15:14		
SC_BOTTOM	29:16	0x0	Bottom scissor
(reserved)	31:30		

SC_BOTTOM_RIGHT: Destination BOTTOM_RIGHT

SC_TOP_LEFT_C MMR:0x1C88 MMR_1:0x1C88 IND:0x1C88 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_LEFT	13:0	0x0	See same field in register SC_TOP_LEFT
(reserved)	15:14		
SC_TOP	29:16	0x0	See same field in register SC_TOP_LEFT
(reserved)	31:30		

SC_TOP_LEFT_C: Aliased to SC_TOP_LEFT

SC_BOTTOM_RIGHT_C MMR:0x1C8C MMR_1:0x1C8C IND:0x1C8C [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SC_RIGHT	13:0	0x0	Right scissor
(reserved)	15:14		

SC_BOTTOM	29:16	0x0	Bottom scissor
(reserved)	31:30		

SC_BOTTOM_RIGHT_C: Aliased to SC_BOTTOM_RIGHT

3.29 Datapath Registers

<No description>

DP_BRUSH_BKGD_CLR MMR:0x1478 MMR_1:0x1478 IND:0x1478 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DP_BRUSH_BKGD_CLR	31:0	0x0	Background color.

DP_BRUSH_BKGD_CLR: <No Description>

DP_BRUSH_FRGD_CLR MMR:0x147C MMR_1:0x147C IND:0x147C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DP_BRUSH_FRGD_CLR	31:0	0x0	Foreground color.

DP_BRUSH_FRGD_CLR: <No Description>

DP_SRC_FRGD_CLR MMR:0x15D8 MMR_1:0x15D8 IND:0x15D8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DP_SRC_FRGD_CLR	31:0	0x0	Foreground color. When color compare src eq flip is enabled, a '1' in bit location n means enable flipping on bit n.

DP_SRC_FRGD_CLR: <No Description>

DP_SRC_BKGD_CLR MMR:0x15DC MMR_1:0x15DC IND:0x15DC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DP_SRC_BKGD_CLR	31:0	0x0	Background color.

DP_SRC_BKGD_CLR: <No Description>

DP_CNTL MMR:0x16C0 MMR_1:0x16C0 IND:0x16C0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_X_DIR	0	0x0	Destination X direction. This bit is written during setup engine initiated operations. This bit is set to '1' by a GUI_MASTER_CNTL write. 0=right to left 1=left to right
DST_Y_DIR	1	0x0	Destination Y direction. 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 0=bottom to top 1=top to bottom
DST_Y_MAJOR	2	0x0	Destination Y major axis flag for bresenham lines. This bit is written during setup engine initiated operations. This bit is assumed to be '1' for all triangles. 0=X major line 1=Y Major line
DST_X_TILE	3	0x0	Enables rectangular tiling in the X direction 0=rectangular tiling in the X direction disabled 1=rectangular tiling in the X direction enabled
DST_Y_TILE	4	0x0	Enables rectangular tiling in the Y direction 0=rectangular tiling in the Y direction disabled 1=rectangular tiling in the Y direction enabled
DST_LAST_PEL	5	0x0	Destination last pel enable for lines. This bit is written during Setup engine operations 0=Destination last pel disabled 1=Destinatino last pel enabled

TRAIL_X_DIR	6	0x0	Trapezoid trailing edge direction. This bit is written during setup engine initiated operations 0=right to left 1=left to right
TRAIL_FILL_DIR	7	0x0	Trapezoid fill direction. 0 = right to left (trailing edge is to the left of the leading edge); 1 = left to right (trailing edge is to the right of the leading edge). This bit is written during setup engine initiated operations. 0=right to left 1=left to right
BRES_SIGN	8	0x0	Bresenham sign. For Trapezoids with sub-pixel addressing, this bit is changed to include pixels on the top/left of the triangle. This bit is automatically set during setup engine operations. 0=Zero error term is positive 1=Zero error term is positive 1. X Major lines and Y_DIR is 0 2. Y Major lines and X_DIR is 0
DP_CULL_TRI	9	0x0	If set, the edgewalker will issue a single, 0 length span and exit. Also denotes that DP_POLY_EDGE should not mask out Z Writes. This bit is written by the Setup engine. 0=Draw triangles normally 1=Draw no pixels for triangles
(reserved)	14:10		
POLY_LINE	15	0x0	Indicates whether the current line is not the last line of a poly line. This bit implies BRUSH tiling. This bit is written during Setup engine operations. This bit is written to '1' by a DP_GUI_MASTER_CNTL write 0=Last or independent line 1=Non-last line of polyline

DP_RASTER_STALL	16	0x0	If set, stall all DST operations until either: a) The Raster has passed the current destination location or b) No Display Offset writes are pending 0=Raster stall disabled 1=Raster stall enabled
DP_TRI_DIS	17	0x0	If set, the edgewalker will accept a triangle from the setup engine, but only issue a single span, representing no pixels. Also denotes that DP_POLY_EDGE should not mask out Z writes. 0=Draw triangles normally 1=Draw no pixels for triangles
DP_POLY_EDGE	18	0x0	Denotes that the line to be drawn is an anti-aliased edge of a polygon. Sub-pixel adjust to first pixel center in the direction of the line, and always mask out Z writes. Always draw last pixel of the line. This bit only applies to 3D texture and shading operations. This bit is written by the Setup engine.
ANTI_ALIAS_INV_DMAJOR	22:19	0x0	Mantissa of the inverse of DMAJOR in normalized (1.xxxx) format. This field is written be the Setup engine.
ANTI_ALIAS_SHIFT	27:23	0x0	Number of right shifts to do to the Bresenham Error term for anti-aliased lines to to produce an error term between 0 and 15. This should be programmed with (1 - (exponent of the inverse of DMAJOR)). This field is written by the Setup engine.
ANTI_ALIAS_SLOPE	31:28	0x0	MSBs of absolute value of the slope. (DMINOR/DMAJOR) 0xF represents 45 degrees. This field is written by the setup engine.

DP_CNTL: <No Description>

DP_DATATYPE MMR:0x16C4 MMR_1:0x16C4 IND:0x16C4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DP_DST_DATATYPE	3:0	0x0	Destination datapath pixel width. Note: choices 7-15 only valid in 3D mode. 2=8 bpp pseudocolor 3=16 bpp aRGB 1555 4=16 bpp RGB 565 5=24 bpp RGB 6=32 aRGB 8888 7=8 bpp RGB 332 8=Y8 greyscale 9=RGB8 greyscale (8 bit intensity, duplicated for all 4 channels. Redchannel is used on writes) 11=YUV 422 packed (VYUY) 12=YUV 422 packed(YVYU) 14=aYUV 444(8:8:8:8) 15=aRGB4444 (intermediate format only. Not understood by the Display Controller)
(reserved)	7:4		

DP_BRUSH_DATATYPE	11:8	0x0	Brush datapath pixel type: 0=8X8 mono pattern (expanded to frgd, bkgd) 1=8X8 mono pattern (expanded to frgd, leave_alone) 2=8X1 mono pattern (expanded to frgd, leave_alone) 3=8X1 mono pattern (expanded to frgd,leave_alone) 4=1X8 mono pattern (expanded to frgd, bkgd) 5=1X8 mono pattern for line (expanded to frgd,leave_alone) 6=32X1 mono pattern for lines (expanded to frgd, bkgd) 7=32X1 mono pattern for line (expanded to frgd,leave_alone) 8=32X32 mono pattern for OPEN GL support (expanded to frgd,bkgd) 9=32X32 mono pattern for OPEN GL support (expanded to frgd,leave_alone) 10=8X8 color (pixel type same as DST) 11=8X1 color (pixel type same as DST) 12=1X8 color (pixel type same as DST) 13=solid color (use frgd) 15=Reserved for ProMo4 Parser. Must not be used by anyone else. Treat as 13, but really means no brush data is to be used
(reserved)	15:12		
DP_SRC_DATATYPE	17:16	0x0	Source datapath pixel type. (If 3D/Scalar operations are in progress, this field is ignored and assumed to be 3). 0=mono (expanded to frgd, bkgd) 1=mono (expanded to frgd, leave_alone) 3=color (pixel type same as DST)
(reserved)	28:18		
HOST_BIG_ENDIAN_EN	29	0x0	Enables big endian data translation for 15 bpp, 16 bpp, and 32 bpp pixel width. In 15 bpp and 16 bpp modes the bytes within each word are swapped. In 32 bpp mode the order of the four bytes within each dword is reversed. 0=big endian datatranslation disabled 1=big endian data translation enabled

DP_BYTE_PIX_ORDER	30	0x0	Reverses the pixel order within each byte in monochrome modes: 0=pixel order from MSBit to LSBit 1=pixel order from LSBit to MSBit
DP_CONVERSION_TEMP	31	0x0	YUV to RGB conversion temperature: 0=red@6500 K, GB@9300 K 1=RGB@9300K

DP_DATATYPE: <No Description>

DP_CNTL_XDIR_YDIR_YMAJOR MMR:0x16D0 MMR_1:0x16D0 IND:0x16D0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	1:0		
DST_Y_MAJOR	2	0x0	Destination Y major axis flag for bresenham lines: 0 = X major line; 1 = Y major line. Note: Can we eliminate this bit and assume everything will be draw Y major? NO. accuracy problem in polylines. 0=X major line 1=Y Major line
(reserved)	14:3		
DST_Y_DIR	15	0x0	Destination Y direction. 0 = bottom to top. 1 = top to bottom 0=bottom to top 1=top to bottom
(reserved)	30:16		
DST_X_DIR	31	0x0	Destination X direction. 0 = right to left. 1 = left to right 0=right to left 1=left to right

DP_CNTL_XDIR_YDIR_YMAJOR: <No Description>

DP_MIX MMR:0x16C8 MMR_1:0x16C8 IND:0x16C8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	7:0		
DP_SRC_SOURCE	10:8	0x0	Src source. Note that during 3D/Scalar Operations (whenever SCALE_3D_FCN is non-zero) the DP_SRC_SOURCE field is ignored and data is always loaded from the 3D/Scalar pipeline 2=loaded from memory (rectangular trajectory 3=loaded thru hostadata (linear trajectory) 4=loaded thru hosdata (linear trajectory & byte-aligned)
(reserved)	15:11		
DP_ROP3	23:16	0x0	Windows 3.1 ROP3 code 0=ROP3 function
(reserved)	31:24		

DP_MIX: <No Description>

DP_WRITE_MSK MMR:0x16CC MMR_1:0x16CC IND:0x16CC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DP_WRITE_MSK	31:0	0x0	Write mask

DP_WRITE_MSK: <No Description>

DP_GUI_MASTER_CNTL MMR:0x146C MMR_1:0x146C IND:0x146C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

GMC_SRC_PITCH_OFFSET_CNTL	0	0x0	Control of SRC_OFFSET, SRC_PITCH 0 = SRC_OFFSET = DEFAULT_OFFSET, SRC_PITCH = DEFAULT_PITCH 1 = leave alone 0=SRC_OFFSET=DEFAULT_OFFSET, SRC_PITCH=DEFAULT_PITCH 1=Leave Alone
GMC_DST_PITCH_OFFSET_CNTL	1	0x0	Control of DST_OFFSET, DST_PITCH: 0 = DST_OFFSET = DEFAULT_OFFSET, DST_PITCH = DEFAULT_PITCH 1 = leave alone 0=DST_OFFSET=DEFAULT_OFFSET, DST_PITCH=DEFAULT_PITCH 1=Leave Alone
GMC_SRC_CLIPPING	2	0x0	Control of SRC scissors: 0 = (SRC_SC_LEFT, SRC_SC_RIGHT) = (DEFAULT_SC_BOTTOM_RIGHT) 1 = leave alone 0=(SRC_SC_RIGHT, SRC_SC_BOTTOM) = (DEFAULT_SC_BOTTOM_RIGHT) 1=no default
GMC_DST_CLIPPING	3	0x0	Control of DST scissors: 0 = (DST_SC_LEFT, DST_SC_RIGHT) = (DEFAULT_SC_BOTTOM_RIGHT) 1 = leave alone 0=(SC_LEFT, SC_TOP) = (0,0), (SC_BOTTOM, SC_RIGHT) = DEF_SC_BOTTOM_RIGHT) 1=no default

GMC_BRUSH_DATATYPE	7:4	0x0	Brush type to use: See DP_BRUSH_DATATYPE in DP_DATATYPE 0=8X8 mono pattern (expanded to frgd, bkgd) 1=8X8 mono pattern (expanded to frgd, leave_alone) 2=8X1 mono pattern (expanded to frgd, leave_alone) 3=8X1 mono pattern (expanded to frgd,leave_alone) 4=1X8 mono pattern (expanded to frgd, bkgd) 5=1X8 mono pattern for line (expanded to frgd,leave_alone) 6=32X1 mono pattern for lines (expanded to frgd, bkgd) 7=32X1 mono pattern for line (expanded to frgd,leave_alone) 8=32X32 mono pattern for OPEN GL support (expanded to frgd,bkgd) 9=32X32 mono pattern for OPEN GL support (expanded to frgd,leave_alone) 10=8X8 color (pixel type same as DST) 11=8X1 color (pixel type same as DST) 12=1X8 color (pixel type same as DST) 13=solid color (use frgd) 15=Reserved for ProMo4 Parser. Must not be used by anyone else. Treat as 13, but really means no brush data is to be used
GMC_DST_DATATYPE	11:8	0x0	Dst type to use: See DP_DST_DATATYPE in DP_DATATYPE 2=8 bpp pseudocolor 3=16 bpp aRGB 1555 4=16 bpp RGB 565 5=24 bpp RGB 6=32 aRGB 8888 7=8 bpp RGB 332 8=Y8 greyscale 9=RGB8 greyscale (8 bit intensity, duplicated for all 4 channels. Redchannel is used on writes) 11=YUV 422 packed (VYUY) 12=YUV 422 packed(YVYU) 14=aYUV 444(8:8:8:8) 15=aRGB4444 (intermediate format only. Not understood by the Display Controller)

GMC_SRC_DATATYPE	13:12	0x0	Src type to use: See DP_SRC_DATATYPE in DP_DATATYPE 0=mono (expanded to frgd, bkgd) 1=mono (expanded to frgd, leave_alone) 3=color (pixel type same as DST)
GMC_BYTE_PIX_ORDER	14	0x0	Mapped to DP_BYTE_PIX_ORDER in DP_DATATYPE 0=pixel order from MSBit to LSBit 1=pixel order from LSBit to MSBit
GMC_CONVERSION_TEMP	15	0x0	Mapped to DP_CONVERSION_TEMP in DP_DATATYPE 0=red@6500 K, GB@9300 K 1=RGB@9300K
GMC_ROP3	23:16	0x0	Mapped to DP_ROP3 in DP_MIX 0=ROP3 function
DP_SRC_SOURCE	26:24	0x0	Mapped to DP_SRC_SOURCE in DP_MIX 2=loaded from memory (rectangular trajectory 3=loaded thru hostadata (linear trajectory) 4=loaded thru hosdata (linear trajectory & byte-aligned)
GMC_3D_FCN_EN	27	0x0	0 = clear SCALE_3D_FCN, Z_EN, STENCIL_EN 1 = leave alone 0=clear SCALE_3D_FCN 1=leave SCALE_3D_FCN alone
GMC_CLR_CMP_CNTL_DIS	28	0x0	0 = leave alone 1 = clear CLR_CMP_FCN_DST, CLR_CMP_FCN_SRC
GMC_AUX_CLIP_DIS	29	0x0	0 = leave alone 1 = clear all AUXn_SC_ENB bits
GMC_WR_MSK_DIS	30	0x0	0 = leave alone 1 = set DP_WRITE_MSK/CLR_CMP_MSK to 0xffffffff

GMC_LD_BRUSH_Y_X	31	0x0	<No Description> 0=leave BRUSH_Y_X alone - Note: for promo4 parser only. 1=Initialize BRUSH_Y_X at end of GMC handler - Note: for promo4 parser only.

DP_GUI_MASTER_CNTL: <No Description>

DP_GUI_MASTER_CNTL_C MMR:0x1C84 MMR_1:0x1C84 IND:0x1C84 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
GMC_SRC_PITCH_OFFSET_CNTL	0	0x0	See same field in register DP_GUI_MASTER_CNTL 0=SRC_OFFSET=DEFAULT_OFFSET, SRC_PITCH=DEFAULT_PITCH 1=Leave Alone
GMC_DST_PITCH_OFFSET_CNTL	1	0x0	See same field in register DP_GUI_MASTER_CNTL 0=DST_OFFSET=DEFAULT_OFFSET, DST_PITCH=DEFAULT_PITCH 1=Leave Alone
GMC_SRC_CLIPPING	2	0x0	See same field in register DP_GUI_MASTER_CNTL 0=(SRC_SC_RIGHT, SRC_SC_BOTTOM) = (DEFAULT_SC_BOTTOM_RIGHT) 1=no default
GMC_DST_CLIPPING	3	0x0	See same field in register DP_GUI_MASTER_CNTL 0=(SC_LEFT, SC_TOP) = (0,0), (SC_BOTTOM, SC_RIGHT) = DEF_SC_BOTTOM_RIGHT) 1=no default

GMC_BRUSH_DATATYPE	7:4	0x0	See same field in register DP_GUI_MASTER_CNTL 0=8X8 mono pattern (expanded to frgd, bkgd) 1=8X8 mono pattern (expanded to frgd, leave_alone) 2=8X1 mono pattern (expanded to frgd, leave_alone) 3=8X1 mono pattern (expanded to frgd,leave_alone) 4=1X8 mono pattern (expanded to frgd, bkgd) 5=1X8 mono pattern for line (expanded to frgd,leave_alone) 6=32X1 mono pattern for lines (expanded to frgd, bkgd) 7=32X1 mono pattern for line (expanded to frgd,leave_alone) 8=32X32 mono pattern for OPEN GL support (expanded to frgd,bkgd) 9=32X32 mono pattern for OPEN GL support (expanded to frgd,leave_alone) 10=8X8 color (pixel type same as DST) 11=8X1 color (pixel type same as DST) 12=1X8 color (pixel type same as DST) 13=solid color (use frgd) 15=Reserved for ProMo4 Parser. Must not be used by anyone else. Treat as 13, but really means no brush data is to be used
GMC_DST_DATATYPE	11:8	0x0	See same field in register DP_GUI_MASTER_CNTL 2=8 bpp pseudocolor 3=16 bpp aRGB 1555 4=16 bpp RGB 565 5=24 bpp RGB 6=32 aRGB 8888 7=8 bpp RGB 332 8=Y8 greyscale 9=RGB8 greyscale (8 bit intensity, duplicated for all 4 channels. Redchannel is used on writes) 11=YUV 422 packed (VYUY) 12=YUV 422 packed(YVYU) 14=aYUV 444(8:8:8:8) 15=aRGB4444 (intermediate format only. Not understood by the Display Controller)
GMC_SRC_DATATYPE	13:12	0x0	See same field in register DP_GUI_MASTER_CNTL 0=mono (expanded to frgd, bkgd) 1=mono (expanded to frgd, leave_alone) 3=color (pixel type same as DST)

GMC_BYTE_PIX_ORDER	14	0x0	See same field in register DP_GUI_MASTER_CNTL 0=pixel order from MSBit to LSBit 1=pixel order from LSBit to MSBit
GMC_CONVERSION_TEMP	15	0x0	See same field in register DP_GUI_MASTER_CNTL 0=red@6500 K, GB@9300 K 1=RGB@9300K
GMC_ROP3	23:16	0x0	See same field in register DP_GUI_MASTER_CNTL 0=ROP3 function
DP_SRC_SOURCE	26:24	0x0	See same field in register DP_GUI_MASTER_CNTL 2=loaded from memory (rectangular trajectory 3=loaded thru hostadata (linear trajectory) 4=loaded thru hosdata (linear trajectory & byte-aligned)
GMC_3D_FCN_EN	27	0x0	See same field in register DP_GUI_MASTER_CNTL 0=clear SCALE_3D_FCN 1=leave SCALE_3D_FCN alone
GMC_CLR_CMP_CNTL_DIS	28	0x0	<No Description>
GMC_AUX_CLIP_DIS	29	0x0	See same field in register DP_GUI_MASTER_CNTL
GMC_WR_MSK_DIS	30	0x0	See same field in register DP_GUI_MASTER_CNTL
(reserved)	31		

DP_GUI_MASTER_CNTL_C: Aliased to register DP_GUI_MASTER_CNTL

3.30 Color Compare Registers

<No description>

CLR_CMP_CLR_SRC MMR:0x15C4 MMR_1:0x15C4 IND:0x15C4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_CLR_SRC	31:0	0x0	Color comparison color of source

CLR_CMP_CLR_SRC: <No Description>

CLR_CMP_CLR_DST MMR:0x15C8 MMR_1:0x15C8 IND:0x15C8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_CLR_DST	31:0	0x0	Color comparison color of destination

CLR_CMP_CLR_DST: <No Description>

CLR_CMP_CNTL MMR:0x15C0 MMR_1:0x15C0 IND:0x15C0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_FN_SRC	2:0	0x0	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) 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)

(reserved)	7:3		
CLR_CMP_FN_DST	10:8	0x0	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_EQ_COLOR, draw when eq) 5 = DST_CLR = CLR_CMP_CLR_DST (CMP_NEQ_COLOR, draw when neq) 6-7 = (reserved) 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)
(reserved)	23:11		
CLR_CMP_SRC	25:24	0x0	Defines source for color keying: 0 = Destination 1 = Source 2 = Src and Dst 3 = (reserved) 0=Destination 1=Source 2=Src and Dst 3=HILITE
(reserved)	31:26		

CLR_CMP_CNTL: <No Description>

CLR_CMP_MSK MMR:0x15CC MMR_1:0x15CC IND:0x15CC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_MSK	31:0	0x0	Color comparison color mask

CLR_CMP_MSK: <No Description>

CLR_CMP_CLR_3D MMR:0x1A24 MMR_1:0x1A24 IND:0x1A24 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

CLR_CMP_CLR_3D	31:0	0x0	Color comparison color.

CLR_CMP_CLR_3D: <No Description>

CLR_CMP_MSK_3D MMR:0x1A28 MMR_1:0x1A28 IND:0x1A28 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CLR_CMP_MSK_3D	31:0	0x0	Color compare mask.

CLR_CMP_MSK_3D: <No Description>

3.31 Parameter FIFO Registers

<No description>

3.32 GUI Engine Control Registers

<No description>

3.33 GUI Engine Status Registers

<No description>

3.34 GUI Bus Mastering Registers

<No description>

3.35 AGP Registers

<No description>

AGP_STATUS CFG:0x0054 MMR:0x0F54 MMR_1:0x0F54 IND:0x0F54 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
RATE1X	0	0x1	<No Description>
RATE2X	1	0x1	<No Description>
RATE4X	2	0x1	<No Description>
(reserved)	8:3		
SBA	9	0x1	<No Description>
(reserved)	23:10		
RQ	31:24	0x1f	<No Description>

AGP_STATUS: <No Description>

AGP_COMMAND CFG:0x0058 MMR:0x0F58 [R] MMR_1:0x0F58 [R] IND:0x0F58 [R] [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
DATA_RATE	2:0	0x0	<No Description>
(reserved)	7:3		
AGP_EN	8	0x0	<No Description> 0=disable 1=enable
SBA_EN (R)	9	0x1	<No Description> 0=Disable 1=Enable
(reserved)	23:10		
RQ_DEPTH	31:24	0x0	<No Description>

AGP_COMMAND: <No Description>

AGP_BASE MMR:0x0170 MMR_1:0x0170 IND:0x0170 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AGP_BASE_ADDR	31:0	0x0	AGP Base Address: NOTE: Bits 21:0 of this field are hardwired to ZERO

AGP_BASE: <No Description>

AGP_CNTL MMR:0x0174 MMR_1:0x0174 IND:0x0174 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AGP_APER_SIZE	5:0	0x0	AGP aperture size 0=000000 = 256MB 32=100000 = 128MB 48=110000 = 64MB 56=111000 = 32MB 60=111100 = 16MB 62=111110 = 8MB 63=111111 = 4MB=63
(reserved)	7:6		
MAX_IDLE_CLK	15:8	0x0	This is the number of clocks (MAX_IDLE_CLK x 32) that the AGP block will wait before stopping the generation of the 2X sideband strobe after it no longer has a request to service.
HOLD_RD_FIFO	16	0x0	<No Description> 0=Normal Operation 1=Hold Fifo
HOLD_RQ_FIFO	17	0x0	<No Description> 0=Normal Operation 1=Hold Fifo

ENABLE_STP_AGP	18	0x0	<No Description> 0=Normal Operation 1=iSTP_AGPb resets AGP block
AGP_OCTWD_ALGN	19	0x0	<No Description> 0=Normal QW Alignment 1=Use OCTWD Alignment
AGP_TG_EXTSENSE	20	0x1	<No Description> 0=Short Fifo Sensing 1=Extended Tag Fifo Sensing
AGP_WRQ_EXTSENSE	21	0x1	<No Description> 0=Short Fifo Sensing 1=Extended Write/Req Fifo Sensing
AGP_RD_EXTSENSE	22	0x1	<No Description> 0=Short Fifo Sensing 1=Extended Read Fifo Sensing
AGP_SOFT_RESET	23	0x0	<No Description> 0=Normal Operation 1=Hold AGP in reset state
RQ_ARB_MAX_CNT	27:24	0x0	<No Description>
RQ_ARB_IDLE_CNT	29:28	0x0	<No Description>
(reserved)	31:30		

AGP_CNTL: <No Description>

AGP_APER_OFFSET MMR:0x0178 MMR_1:0x0178 IND:0x0178 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AGP_APER_OFFSET	25:0	0x2000000	<No Description> NOTE: Bits 24:0 of this field are hardwired to ZERO

(reserved)	31:26		

AGP_APER_OFFSET: <No Description>

3.36 Miscellaneous

Miscellaneous RegistFunction: Overlay RegistersActive Field: OV0_SCALE_Y2R_DIS0x0 : Divide by 10x0 : Divide by 1

PMI_CAP_ID CFG:0x005C MMR:0x0F5C MMR_1:0x0F5C IND:0x0F5C [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PMI_CAP_ID	7:0	0x1	Indicates this capability is the PCI Power Mangement Interface (PMI) register. 1=PCI Bus Power Management Interface (PMI) register section

PMI_CAP_ID: Capability ID

PMI_NXT_CAP_PTR CFG:0x005D MMR:0x0F5D MMR_1:0x0F5D IND:0x0F5D [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PMI_NXT_CAP_PTR	7:0	0x0	0=Last function in capabilities list

PMI_NXT_CAP_PTR: Next capability pointer.

PMI_PMC_REG CFG:0x005E MMR:0x0F5E MMR_1:0x0F5E IND:0x0F5E [R] 16 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PMI_VERSION	2:0	0x2	2=Compliant with PMI Specification verison 1.1
PMI_PME_CLOCK	3	0x0	0 = No PCI clock needed to generate PME#. Function can not assert PME#.
(reserved)	4		
PMI_DEV_SPECIFIC_INIT	5	0x0	0 = Device specific initialization not needed for this device.
(reserved)	8:6		
PMI_D1_SUPPORT	9	0x1	1 = Power state D1 (standby) supported by this device.

PMI_D2_SUPPORT	10	0x0	0 = Power state D2 (suspend) not supported by this device.
PMI_PME_SUPPORT	15:11	0x0	00000 = Device can not assert PME# from any power state.

PMI_PMC_REG: PCI PMI Power Management Capabilities (PMC)

PMI_PMCSR_REG CFG:0x0060 MMR:0x0F60 [R] MMR_1:0x0F60 [R] IND:0x0F60 [R] [RW] 16 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PMI_POWER_STATE	1:0	0x0	Write: Sets device into specified power state. Writes of unsupported states are not accepted. Read: Indicates current power state of the device. 00 = D0 state (on). 01 = D1 state (standby). 10 = D2 state (suspend, not support on Rage 128). 11 = D3 state (off).
(reserved)	7:2		
PMI_PME_EN (R)	8	0x0	0 = Device does not support PME# generation.
PMI_DATA_SELECT (R)	12:9	0x0	Device does not support the PMI data register. Will read back zeros.
PMI_DATA_SCALE (R)	14:13	0x0	Device does not support the PMI data register. Will read back zeros.
PMI_PME_STATUS (R)	15	0x0	0 = Device does not support PME# generation.

PMI_PMCSR_REG: PCI PMI Power Management Control/Status.

PMI_DATA CFG:0x0063 MMR:0x0F63 MMR_1:0x0F63 IND:0x0F63 [R] 8 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PMI_DATA	7:0	0x0	PMI data register not supported in Rage 128. Will read back zeros.

PMI_DATA: PCI PMI data register.

MM_INDEX MMR:0x0000 MMR_1:0x0000 IOR:0x0000 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MM_ADDR	26:0	0x0	<No Description> NOTE: Bits 1:0 of this field are hardwired to ZERO
(reserved)	30:27		
MM_APER	31	0x0	<No Description> 0=Register Aperture 1=Linear Aperture 0

MM_INDEX: <No Description>

MM_DATA MMR:0x0004 MMR_1:0x0004 IOR:0x0004 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MM_DATA	31:0	0x0	<No Description>

MM_DATA: <No Description>

BUS_CNTL1 MMR:0x0034 MMR_1:0x0034 IOR:0x0034 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PMI_IO_DISABLE	0	0x0	<No Description> 0=Normal 1=Disable
PMI_MEM_DISABLE	1	0x0	<No Description> 0=Normal 1=Disable

PMI_BM_DISABLE	2	0x0	<No Description> 0=Normal 1=Disable
PMI_INT_DISABLE	3	0x0	<No Description> 0=Normal 1=Disable
BUS_EN_WAIT_ON_LOCK	4	0x0	<No Description> 0=Normal 1=Enable locked cycle and bus master fix for Triton chip sets
(reserved)	31:5		

BUS_CNTL1: <No Description>

GEN_INT_CNTL MMR:0x0040 MMR_1:0x0040 IOR:0x0040 IND:0x0040 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_INT_EN	0	0x0	Vertical blank interrupt enable. 0=Disable 1=Enable
CRTC_VLINE_INT_EN	1	0x0	Vertical line interrupt enable. 0=Disable 1=Enable
CRTC_VSYNC_INT_EN	2	0x0	Vertical sync interrupt enable. 0=Disable 1=Enable
SNAPSHOT_INT_EN	3	0x0	Snapshot interrupt enable. 0=Disable 1=Enable
(reserved)	9:4		

FP_DETECT_INT_EN	10	0x0	Enables the Flat Panel detection interrupt for panels with hot-plugging support.
(reserved)	15:11		
BUSMASTER_EOL_INT_EN	16	0x0	Bus master end-of-system-list interrupt enable. 0=Disable 1=Enable
I2C_INT_EN	17	0x0	I2C interrupt enable. 0=Disable 1=Enable
MPP_GP_INT_EN	18	0x0	<No Description> 0=Disable 1=Enable
GUI_IDLE_INT_EN	19	0x0	<No Description> 0=Disable 1=Enable
(reserved)	23:20		
VIPH_INT_EN	24	0x0	<No Description> 0=Disable 1=Enable
(reserved)	31:25		

GEN_INT_CNTL: Interrupt enables. Setting bits allows corresponding status bit to generate an interrupt signal to the system. No effect if strapped to interrupt disable.

GEN_INT_STATUS MMR:0x0044 MMR_1:0x0044 IOR:0x0044 IND:0x0044 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_VBLANK_INT (R)	0	0x0	Vertical blank started since last cleared. 0=No event 1=Event has occurred, interrupting if enabled

CRTC_VBLANK_INT_AK (W)	0	0x0	Write '1' clears CRTC_VBLANK_INT status. 0=No effect 1=Clear status
CRTC_VLINE_INT_AK (W)	1	0x0	Write '1' clears CRTC_VLINE_INT status. 0=No effect 1=Clear status
CRTC_VLINE_INT (R)	1	0x0	Vertical line trigger point reached since last cleared. 0=No event 1=Event has occurred, interrupting if enabled
CRTC_VSYNC_INT_AK (W)	2	0x0	Write '1' clears CRTC_VSYNC_INT status. 0=No effect 1=Clear status
CRTC_VSYNC_INT (R)	2	0x0	Vertical sync started since last cleared. 0=No event 1=Event has occurred, interrupting if enabled
SNAPSHOT_INT_AK (W)	3	0x0	Write '1' clears SNAPSHOT_INT status. 0=No effect 1=Clear status
SNAPSHOT_INT (R)	3	0x0	Snapshot taken since last cleared. 0=No event 1=Event has occurred, interrupting if enabled
(reserved)	7:4		
CAP0_INT_ACTIVE (R)	8	0x0	Indicates capture port 0 is generating an interrupt. 0=Capture port 0 not source of any active interrupt 1=Capture port 0 has active interrupt(s)
CAP1_INT_ACTIVE (R)	9	0x0	Indicates capture port 1 is generating an interrupt. 0=Capture port 1 not source of any active interrupt 1=Capture port 1 has active interrupt(s)

FP_DETECT_INT_AK (W)	10	0x0	Write only bit. Clears FP_DETECT_INT status. Need to invert value of FP_DETECT_INT_POL as described in FP_GEN_CNTL register
FP_DETECT_INT (R)	10	0x0	Read only bit. This bit indicates when the event (Flat Panel Connect/Disconnect) specified by FP_DETECT_INT_POL in FP_GEN_CNTL has occurred on a panel with hot-plugging support.
(reserved)	15:11		
BUSMASTER_EOL_INT_AK (W)	16	0x0	<No Description> 0=No effect 1=Clear status
BUSMASTER_EOL_INT (R)	16	0x0	<No Description> 0=No event 1=Event has occurred, interrupting if enabled
I2C_INT_AK (W)	17	0x0	<No Description> 0=No effect 1=Clear status
I2C_INT (R)	17	0x0	<No Description> 0=No event 1=Event has occurred, interrupting if enabled
MPP_GP_INT_AK (W)	18	0x0	<No Description> 0=No effect 1=Clear status
MPP_GP_INT (R)	18	0x0	<No Description> 0=No event 1=Event has occurred, interrupting if enabled
GUI_IDLE_INT_AK (W)	19	0x0	<No Description> 0=No effect 1=Clear status

GUI_IDLE_INT (R)	19	0x1	<No Description> 0=No event 1=Event has occurred, interrupting if enabled
(reserved)	23:20		
VIPH_INT (R)	24	0x0	<No Description> 0=No event 1=Event has occurred, interrupting if enabled
(reserved)	31:25		

GEN_INT_STATUS: Interrupt & Status indicators. Read shows current states. Write of 1 clears states. Note each field may be used for polling, even if not enabled to generate an interrupt.

GPIO_MONID MMR:0x0068 MMR_1:0x0068 IOR:0x0068 IND:0x0068 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MONID_A	3:0	0x0	software controlled out-going 'A' pin of each MONID pad register bit to pin mapping: GPIO_MONID(0) -> MONID0_A GPIO_MONID(1) -> MONID1_A GPIO_MONID(2) -> MONID2_A GPIO_MONID(3) -> MONID3_A These bits must be reset to 0 before the MASK is turned off
(reserved)	7:4		
MONID_Y (R)	11:8	0x0	the 'Y' pins of each pad allow software to read the logic state of each pad GPIO_MONID(5) => MONID0_Y GPIO_MONID(6) => MONID1_Y GPIO_MONID(7) => MONID2_Y GPIO_MONID(8) => MONID3_Y
(reserved)	15:12		

MONID_EN	19:16	0x0	pad direction control 0 = input mode 1 = output mode GPIO_MONID(8) => MONID0_EN GPIO_MONID(9) => MONID1_EN GPIO_MONID(10) => MONID2_EN GPIO_MONID(11) => MONID3_EN These bits must be reset to 0 before the MASK is turned off
(reserved)	23:20		
MONID_MASK	27:24	0x0	0 = gpio turned off 1 = gpio enable GPIO_MONID(12) => MONID0_MASK GPIO_MONID(13) => MONID1_MASK GPIO_MONID(14) => MONID2_MASK GPIO_MONID(15) => MONID3_MASK
(reserved)	31:28		

GPIO_MONID: <No Description>

SEPROM_CNTL MMR:0x006C MMR_1:0x006C IOR:0x006C IND:0x006C [RW] 32 bits (access: 8/16/32)			
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
(reserved)	6		
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
SCK_PRESCALE	10:9	0x0	'00' sets the scale to 16 XCLKS = 1 SCK '01' sets the scale to 8 XCLKS = 1 SCK '10' sets the scale to 32 XCLKS = 1 SCK '11' sets the scale to 64 XCLKS = 1 SCK
BCNT_OVER_WTE_EN	11	0x1	This bit must be set to '1' for burst ROM write/read. This bit works coherently with the BYTE_CNT.
RB_MASKB	12	0x0	<No Description>
(reserved)	15:13		
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
(reserved)	31:24		

SEPROM_CNTL: <No Description>

AMCGPIO_MASK_MIR MMR:0x009C MMR_1:0x009C IOR:0x009C IND:0x009C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_MASK	31:0	0x0	<No Description>

AMCGPIO_MASK_MIR: <No Description>

AMCGPIO_A_MIR MMR:0x00A0 MMR_1:0x00A0 IOR:0x00A0 IND:0x00A0 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_A	31:0	0x0	<No Description>

AMCGPIO_A_MIR: <No Description>

AMCGPIO_Y_MIR MMR:0x00A4 MMR_1:0x00A4 IOR:0x00A4 IND:0x00A4 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_Y (R)	31:0	0x0	<No Description>

AMCGPIO_Y_MIR: <No Description>

AMCGPIO_EN_MIR MMR:0x00A8 MMR_1:0x00A8 IOR:0x00A8 IND:0x00A8 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_EN	31:0	0x0	<No Description>

AMCGPIO_EN_MIR: <No Description>

PALETTE_INDEX MMR:0x00B0 MMR_1:0x00B0 IOR:0x00B0 IND:0x00B0 [RW] 32 bits (access: 8/16/32)			
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.
(reserved)	15:8		
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.
(reserved)	31:24		

PALETTE_INDEX: Display palette read and write index setting.
Recommend using byte writes to set either read mode or write mode for the palette.

PALETTE_DATA MMR:0x00B4 MMR_1:0x00B4 IOR:0x00B4 IND:0x00B4 [RW] 32 bits (access: 8/16/32)			
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.
(reserved)	31:24		

PALETTE_DATA: Display palette data read/write.

GEN_RESET_CNTL MMR:0x00F0 MMR_1:0x00F0 IOR:0x00F0 IND:0x00F0 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SOFT_RESET_GUI	0	0x0	<No Description> 0=Not reset 1=Reset
(reserved)	7:1		
SOFT_RESET_VCLK	8	0x0	Resets pixel clock based logic. 0=Not reset 1=Reset
SOFT_RESET_PCLK	9	0x0	Resets character clock based logic. 0=Not reset 1=Reset
(reserved)	10		
SOFT_RESET_DISPENG_XCLK	11	0x0	Resets display logic on memory clock. 0=Not reset 1=Reset

SOFT_RESET_MEMCTLR_XCLK	12	0x0	Resets memory controller logic. 0=Not reset 1=Reset
(reserved)	31:13		

GEN_RESET_CNTL: Soft reset controls for various blocks.

GEN_STATUS MMR:0x00F4 MMR_1:0x00F4 IOR:0x00F4 IND:0x00F4 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	0		
MPP_GP_RDY (R)	1	0x0	Read only bit; '0' means the MPPGP bus is idle; '1' means the MPPGP bus is busy 0=MPP_BUS_RDY 1=MPP_BUS_BUSY
MPP_GP_ALT_RDY (R)	2	0x0	Read only bit; This bit tells the status of the MPP_ALT_REG statemachine '0' means the MPP_ALT_REG statemachine is not ready; '1' means the MPP_ALT_REG statemachine is ready, and the MPP_GP_ALT_REG_ADDR and MPP_GP_ALT_REG_DATA can be programmed respectively
(reserved)	3		
MPP_GP_INT_FLAG (R)	4	0x0	Read only bit; This is the mirror of the MPP_GP_INT field in the GEN_INT_STATUS 0=No event 1=Event has occurred, interrupting if enabled
(reserved)	31:5		

GEN_STATUS: Various status indicators.

CONFIG_APER_SIZE MMR:0x0108 MMR_1:0x0108 IND:0x0108 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
APER_SIZE	25:0	0x2000000	Size of linear apertures (both 0 and 1). This includes both the frame buffer image and the AGP system memory image area. NOTE: Bits 24:0 of this field are hardwired to ZERO
(reserved)	31:26		

CONFIG_APER_SIZE: Linear Aperture Size

CONFIG_REG_1_BASE MMR:0x010C MMR_1:0x010C IND:0x010C [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	12:0		
REG_1_BASE	31:13	0x0	Base address of register aperture 1. The base address of register aperture 0 is found in PCI configuration space. The first and second register apertures are indentical. The second is intended for use in PowerMac systems, but functions in all systems. NOTE: Bit 0 of this field is hardwired to ONE

CONFIG_REG_1_BASE: Register Aperture 1 Base

CONFIG_REG_APER_SIZE MMR:0x0110 MMR_1:0x0110 IND:0x0110 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
REG_APER_SIZE	13:0	0x2000	Size in bytes of each of the register apertures (both 0 and 1). NOTE: Bits 12:0 of this field are hardwired to ZERO
(reserved)	31:14		

CONFIG_REG_APER_SIZE: Register Aperture Size

CONFIG_MEMSIZE_EMBEDDED MMR:0x0114 MMR_1:0x0114 IND:0x0114 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CONFIG_MEMSIZE_EMB	25:0	0x0	Reserved for future use. This will indicate the size in bytes of the on-chip portion of the frame buffer.
(reserved)	31:26		

CONFIG_MEMSIZE_EMBEDDED: Embedded Memory Size

HW_DEBUG2 MMR:0x011C MMR_1:0x011C IND:0x011C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
HW_20_DEBUG	0	0x0	<No Description>
HW_21_DEBUG	1	0x0	<No Description>
HW_22_DEBUG	2	0x0	<No Description>
HW_23_DEBUG	3	0x0	<No Description>
HW_24_DEBUG	4	0x0	<No Description>
HW_25_DEBUG	5	0x0	<No Description>
HW_26_DEBUG	6	0x0	<No Description>
HW_27_DEBUG	7	0x0	<No Description>
HW_28_DEBUG	8	0x0	<No Description>
HW_29_DEBUG	9	0x0	<No Description>
HW_2A_DEBUG	10	0x0	<No Description>
HW_2B_DEBUG	11	0x0	<No Description>
HW_2C_DEBUG	12	0x0	<No Description>
HW_2D_DEBUG	13	0x0	<No Description>

HW_2E_DEBUG	14	0x0	<No Description>
HW_2F_DEBUG	15	0x0	<No Description>
HW_30_DEBUG	16	0x0	<No Description>
HW_31_DEBUG	17	0x0	<No Description>
HW_32_DEBUG	18	0x0	<No Description>
HW_33_DEBUG	19	0x0	<No Description>
HW_34_DEBUG	20	0x0	<No Description>
HW_35_DEBUG	21	0x0	<No Description>
HW_36_DEBUG	22	0x0	<No Description>
HW_37_DEBUG	23	0x0	<No Description>
HW_38_DEBUG	24	0x0	<No Description>
HW_39_DEBUG	25	0x0	<No Description>
HW_3A_DEBUG	26	0x0	<No Description>
HW_3B_DEBUG	27	0x0	<No Description>
HW_3C_DEBUG	28	0x0	<No Description>
HW_3D_DEBUG	29	0x0	<No Description>
HW_3E_DEBUG	30	0x0	<No Description>
HW_3F_DEBUG	31	0x0	<No Description>

HW_DEBUG2: <No Description>

TEST_DEBUG_OUT MMR:0x012C MMR_1:0x012C IND:0x012C [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

TEST_DEBUG_OUTR	10:0	0x0	Allows read-back of the current state of the TEST_DEBUG_OUT bus. Since register reads are multi-cycle events, the value read can only be considered valid if the value on the bus is static.
(reserved)	31:11		

TEST_DEBUG_OUT: Test&Debug Output

SW_SEMAPHORE MMR:0x013C MMR_1:0x013C IND:0x013C [RW] 32 bits (access: 8/16/32)			
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.
(reserved)	31:16		

SW_SEMAPHORE: Scratch register.

MEM_SDRAM_MODE_REG MMR:0x0158 MMR_1:0x0158 IND:0x0158 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MEM_MODE_REG	13:0	0x0	Value programmed into SDRAM mode register when SDRAM reset sequence is initiated.
(reserved)	15:14		
MEM_BURST_LENGTH	18:16	0x0	SDRAM burst length. 1=2 2=4 3=8
MEM_BURST_MODE	19	0x0	SDRAM burst mode. 0=Sequential 1=Interleaved
MEM_CAS_LATENCY	22:20	0x3	SDRAM CAS Latency. 2=2 3=3
(reserved)	30:23		

MEM_SDRAM_RESET	31	0x0	Initiate SDRAM reset sequence on a 0 to 1 transition of this register bit. 0=Normal 1=Reset

MEM_SDRAM_MODE_REG: SDRAM Mode Register Control.

PAD_AGPINPUT_DELAY MMR:0x0164 MMR_1:0x0164 IND:0x0164 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PAD_AGPINPUT_DELAY	31:0	0x0	<No Description>

PAD_AGPINPUT_DELAY: <No Description>

PAD_CTLR_STRENGTH MMR:0x0168 MMR_1:0x0168 IND:0x0168 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PAD_N_STRENGTH_READ_BACK (R)	3:0	0x0	<No Description>
PAD_P_STRENGTH_READ_BACK (R)	7:4	0x0	<No Description>
PAD_N_MANUAL_STRENGTH	11:8	STRAP	<No Description>
PAD_P_MANUAL_STRENGTH	15:12	STRAP	<No Description>
PAD_MANUAL_OVERRIDE	16	0x1	<No Description> 0=Allow normal impedance compensation operation 1=Default to manual settings
PAD_TEST_OUT (R)	17	0x0	<No Description>
PAD_DUMMY_OUT (R)	18	0x0	<No Description>
(reserved)	19		
PAD_HON	20	0x0	<No Description> 0=Agp pads hysteresis off 1=Agp pads hysteresis on

(reserved)	31:21		

PAD_CTLR_STRENGTH: <No Description>

PAD_CTLR_UPDATE MMR:0x016C MMR_1:0x016C IND:0x016C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PAD_UPDATE_RATE	4:0	0x16	<No Description>
(reserved)	7:5		
PAD_SAMPLE_DELAY	12:8	0x6	<No Description>
(reserved)	15:13		
PAD_INC_THRESHOLD	20:16	0x18	<No Description>
(reserved)	23:21		
PAD_DEC_THRESHOLD	28:24	0x8	<No Description>
(reserved)	31:29		

PAD_CTLR_UPDATE: <No Description>

PCI_GART_PAGE MMR:0x017C MMR_1:0x017C IND:0x017C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
PCI_GART_DIS	0	0x1	<No Description> 0=Enable 1=Disable Note: PCI_GART_DIS =1 is required if AGP is used. 0=Enable 1=Disable
(reserved)	11:1		

PCI_GART_PAGE	31:12	0x0	This is a 32bit physical memory address to a 32KB table of page entries. The table has the following form : DWORD PhysPageNo[8192] When in PCI mode and Promo4, BusMastering is enabled, any reference to AGP offsets now uses this paging mechanism to reference physical memory. Given any 32bit physical 'AGP-offset' , [24:12] is the pageIndex. PhysPageNo[pageIndex] is the host memory physical page number. Actual referenced address is : PhysAddr[31:12]<-PhysPageNo PhysAddr[11:0]<-Offset[11:0]. PhysAddr is then used as the address of a PCI busmastering read.

PCI_GART_PAGE: With PCI_GART_PAGE, you can get up to a 32MB contiguous address space into PCI system memory via a scatter gather mechanism.

VIDEOMUX_CNTL MMR:0x0190 MMR_1:0x0190 IND:0x0190
[RW] 32 bits (access: 8/16/32)

Field Name	Bits	Default	Description
VIDEO_EN_LOW	0	0x1	1 = pdata port used for video capture default = 0
VIDEO_EN_HIGH	1	0x1	1 = part of memory bus (MD[70:79]) is used as extension to video capture 0 = default
LCD_I2C	2	0x0	<No Description>
(reserved)	3		
AMCGPIO_GA_DRV	4	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(7:0)
AMCGPIO_GB_DRV	5	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(11:8)

AMCGPIO_GC_DRV	6	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(20:12)
AMCGPIO_GD_DRV	7	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(21)
AMCGPIO_GE_DRV	8	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(23:22)
AMCGPIO_GF_DRV	9	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(25:24)
AMCGPIO_GG_DRV	10	0x1	<No Description> 0=supply 4mA 1=supply 8mA to amcgpio(27:26)
(reserved)	15:11		
ROM_CLK_DIVIDE	20:16	0x5	romclk_divide is used adjust rom timing to big range of xclk frequency.
STR_ROMCLK	21	0x0	<No Description>
(reserved)	31:22		

VIDEOMUX_CNTL: <No Description>

AMCGPIO_MASK MMR:0x0194 MMR_1:0x0194 IND:0x0194 **[RW] 32 bits (access: 8/16/32)**			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_MASK	31:0	0x0	Each bit in this register makes the same bit in the AMCGPIO_A_REG and AMCGPIO_EN_REG effective

AMCGPIO_MASK: <No Description>

MDGPIO_MASK MMR:0x0198 MMR_1:0x0198 IND:0x0198 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MDGPIO_MASK	31:0	0x0	Each bit in this register makes the same bit in the MDGPIO_A_REG and MDGPIO_EN_REG effective

MDGPIO_MASK: <No Description>

AMCGPIO_A_REG MMR:0x01A0 MMR_1:0x01A0 IND:0x01A0 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_A	31:0	0x0	This register controls the 'a' pin of 26 pads, but each bit inside this register is effective only if the the same bit inside the AMCGPIO_MASK is turned on. These bits must be reset to 0 before the MASK is turned off. The register bits are mapped to pins as follows: AMCGPIO_A(3:0) Address/Data for MPP AMCGPIO_A(7:4) Address/Data for MPP or VIP HAD(7:4) AMCGPIO_A(8) MPP AS or VIPCLK AMCGPIO_A(9) MPP DS or VIP HCTL AMCGPIO_A(10) MPP SRDY or VIP HAD(0) AMCGPIO_A(11) BUS_CLK_SEL_STRAP / Clock for eprom flops / VIP HAD(1) AMCGPIO_A(19:12) DVS data in AMCGPIO_A(20) DVS clock in AMCGPIO_A(21) BYTCLK AMCGPIO_A(22) I2C SDA / VIP HAD(2) AMCGPIO_A(23) I2C SCL / VIP HAD(3) / VIP interrupt AMCGPIO_A(24) LCDCLK AMCGPIO_A(25) LCDCDE

AMCGPIO_A_REG: <No Description>

AMCGPIO_Y_REG MMR:0x01A4 MMR_1:0x01A4 IND:0x01A4 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_Y (R)	31:0	0x0	Reading from this register gives the logic value on the 'p' pin of the corresponding pad

AMCGPIO_Y_REG: <No Description>

AMCGPIO_EN_REG MMR:0x01A8 MMR_1:0x01A8 IND:0x01A8 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AMCGPIO_EN	31:0	0x0	This register controls the output enable of 26 pads, but each bit inside this register is effective only if the the same bit inside the AMCGPIO_MASK is turned on. This register must be reset to 0 before the MASK is turned off. Turning on the enable will make that pad an output from the chip, turning it off makes that pad input The register bits are mapped to pins as follows: AMCGPIO_A(3:0) Address/Data for MPP AMCGPIO_A(7:4) Address/Data for MPP or VIP HAD(7:4) AMCGPIO_A(8) MPP AS or VIPCLK AMCGPIO_A(9) MPP DS or VIP HCTL AMCGPIO_A(10) MPP SRDY or VIP HAD(0) AMCGPIO_A(11) BUS_CLK_SEL_STRAP / Clock for eprom flops / VIP HAD(1) AMCGPIO_A(19:12) DVS data in AMCGPIO_A(20) DVS clock in AMCGPIO_A(21) BYTCLK AMCGPIO_A(22) I2C SDA / VIP HAD(2) AMCGPIO_A(23) I2C SCL / VIP HAD(3) / VIP interrupt AMCGPIO_A(24) LCDCLK AMCGPIO_A(25) LCDCDE

AMCGPIO_EN_REG: <No Description>

MDGPIO_A_REG MMR:0x01AC MMR_1:0x01AC IND:0x01AC [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

MDGPIO_A	31:0	0x0	This register controls the 'a' pin of the DQ pads, but each bit inside this register is effective only if the the same bit inside the MDGPIO_MASK is turned on. These bits must be reset to 0 before the MASK is turned off. These register bits are mapped to pins as follows: MDGPIO_A(7:6) DQ(71:70) ZV Control Port MDGPIO_A(15:8) DQ(79:72) Extended VIP/DVS port/ZV data in MDGPIO_A(22) DQ(86) DS for MPP2/I2C SDA MDGPIO_A(23) DQ(87) AS for MPP2/I2C SCL MDGPIO_A(31:24) DQ(95:88) Address/Data for MPP2

MDGPIO_A_REG: <No Description>

MDGPIO_EN_REG MMR:0x01B0 MMR_1:0x01B0 IND:0x01B0 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MDGPIO_EN	31:0	0x0	Turning on the enable will make that pad an output from the chip, turning it off makes that pad input. These bits must be reset before the MASK is turned off.

MDGPIO_EN_REG: <No Description>

MDGPIO_Y_REG MMR:0x01B4 MMR_1:0x01B4 IND:0x01B4 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
MDGPIO_Y (R)	31:0	0x0	Reading from this register gives the logic value on the 'p' pin of the corresponding pad

MDGPIO_Y_REG: <No Description>

VIPH_CH0_DATA MMR:0x0C00 MMR_1:0x0C00 IND:0x0C00 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

VIPH_CH0_DT	31:0	0x0	<No Description>

VIPH_CH0_DATA: <No Description>

VIPH_CH1_DATA MMR:0x0C04 MMR_1:0x0C04 IND:0x0C04 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VIPH_CH1_DT	31:0	0x0	<No Description>

VIPH_CH1_DATA: <No Description>

VIPH_CH2_DATA MMR:0x0C08 MMR_1:0x0C08 IND:0x0C08 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VIPH_CH2_DT	31:0	0x0	<No Description>

VIPH_CH2_DATA: <No Description>

VIPH_CH3_DATA MMR:0x0C0C MMR_1:0x0C0C IND:0x0C0C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
VIPH_CH3_DT	31:0	0x0	<No Description>

VIPH_CH3_DATA: <No Description>

CRTC_CRNT_FRAME MMR:0x0214 MMR_1:0x0214 IND:0x0214 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CRTC_CRNT_FRAME (R)	20:0	0x0	Readback of current value of display frame counter. Used by display time sensitive applications such as video playback.
(reserved)	31:21		

CRTC_CRNT_FRAME: Current Frame

SNAPSHOT_VH_COUNTS MMR:0x0240 MMR_1:0x0240 IND:0x0240 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SNAPSHOT_HCOUNT	8:0	0x0	Snapshot of CRTC vertical count value.
(reserved)	15:9		
SNAPSHOT_VCOUNT	26:16	0x0	Snapshot of CRTC horizontal count value.
(reserved)	31:27		

SNAPSHOT_VH_COUNTS: <No Description>

SNAPSHOT_F_COUNT MMR:0x0244 MMR_1:0x0244 IND:0x0244 [R] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
SNAPSHOT_F_COUNT	20:0	0x0	Snapshot of CRTC frame count value.
(reserved)	31:21		

SNAPSHOT_F_COUNT: <No Description>

N_VIF_COUNT MMR:0x0248 MMR_1:0x0248 IND:0x0248 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
N_VIF_COUNT_VAL	9:0	0x0	Programmable N-video-in-field count value which is used to generate a snapshot interrupt when this N-count value is equal to the count value of the lower 10-bit SNAPSHOT_VIF_COUNT (please also refer to CRTC_INT_CNTL register[8:7] - 0_06 for the snapshot interrupt specification).
(reserved)	30:10		
GENLOCK_SOURCE_SEL	31	0x0	<No Description>

N_VIF_COUNT: <No Description>

SNAPSHOT_VIF_COUNT MMR:0x024C MMR_1:0x024C IND:0x024C [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
LSNAPSHOT_VIF_COUNT (R)	9:0	0x0	Lower Snapshot of Video-in-field count value.(Lower 10-bit [9:0] indicates the current number of frames
USNAPSHOT_VIF_COUNT (R)	20:10	0x0	Upper Snapshot of Video-in-field count value.(Upper 11-bit [20:10] indicates the number of N-frames) - Also see Note 1.
(reserved)	23:21		
AUTO_SNAPSHOT_TAKEN_WR (W)	24	0x0	<No Description>
AUTO_SNAPSHOT_TAKEN_RD (R)	24	0x0	<No Description>
MANUAL_SNAPSHOT_NOW	25	0x0	1 = Snapshot taken immediately(writing ‘1’ to this bit prevents all auto-snapshot taking until a write of ‘0’ to the AUTO_SNAPSHOT_TAKEN bit that will re-enable the auto-snapshot taking.)
(reserved)	31:26		

SNAPSHOT_VIF_COUNT: <No Description>

DAC_EXT_CNTL MMR:0x0280 MMR_1:0x0280 IND:0x0280 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	3:0		
DAC_FORCE_BLANK_OFF_EN	4	0x0	Enable Forcing the DAC Blank signal Off 0=Normal DAC BLANK functionality 1=DAC BLANK forced off
DAC_FORCE_DATA_EN	5	0x0	Enable Forcing the RGB inputs to the DAC to the value specified by the DAC_FORCE_DATA and DAC_FORCE_DATA_SEL registers. 1 = Force DAC values to DAC_FORCE_DATA register values 0 = Normal DAC operation 0=Disable 1=Enable
DAC_FORCE_DATA_SEL	7:6	0x0	Selects which of the R/G/B lines to the DAC have the value specified in DAC_FORCE_DATA register 00 - R = DAC_FORCE_DATA / G = 0 / B = 0 01 - R = 0 / G = DAC_FORCE_DATA / B = 0 10 - R = 0 / G = 0 / B = DAC_FORCE_DATA 11 - R = G = B = DAC_FORCE_DATA 0=R=DAC_FORCE_DATA,G=B=? 1=G=DAC_FORCE_DATA,R=B=? 2=B=DAC_FORCE_DATA,R=G=? 3=R=G=B=DAC_FORCE_DATA
DAC_FORCE_DATA	15:8	0x0	Contains the 8-bit value that will be forced onto the DAC R/G/B inputs as determined by DAC_FORCE_DATA_SEL when DAC_FORCE_DATA_EN is set to '1'.
(reserved)	31:16		

DAC_EXT_CNTL: Extended DAC control register

OV0_COL_CONV MMR:0x04FC MMR_1:0x04FC IND:0x04FC [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**

OV0_CB_TO_B	6:0	0x0	<No Description>
(reserved)	7		
OV0_CB_TO_G	15:8	0x0	<No Description>
OV0_CR_TO_G	23:16	0x0	<No Description>
OV0_CR_TO_R	30:24	0x0	<No Description>
OV0_NEW_COL_CONV	31	0x0	<No Description> 0=USE OLD COL_CONV 1=USE NEW COL_CONV

OV0_COL_CONV: <No Description>

PM4_BUFFER_DL_WPTR_DELAY MMR:0x0718 MMR_1:0x0718 IND:0x0718 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PRE_WRITE_TIMER	27:0	0x0	This field specifies the delay from a write to PM4_BUFFER_DL_WPTR_DELAY to when PM4 engine sees that. Default to zero.
PRE_WRITE_LIMIT	31:28	0x0	the number of times allowed to interupt and reset the transfer timer before a force transfer is done. Default to zero.

PM4_BUFFER_DL_WPTR_DELAY: In Rage128 Pro, an extra stage is added to the updating of the PM4_BUFFER_DL_WPTR such that a write to that address actually goes to a 'PRE' register. After a certain programmable delay (specified in this register), the value would get transferred to the original register. This is done as an effort to avoid the HW PM4 engine from fetching data too early from memory. If PM4_BUFFER_DL_WPTR is written again before timeout, the timer gets set and starts again. The original register does not get updated while 'PRE' register is overwritten with the new value. There is a limit on such preemption as an infinite loop is allowed otherwise. After such limit is reached, the next write to PM4_BUFFER_DL_WPTR will force whatever in the 'PRE' register to be transferred to the original register with the new value put in the 'PRE' register.

CRC_CMDFIFO_ADDR MMR:0x0740 MMR_1:0x0740 IND:0x0740 [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CRC_CMDFIFO_ADDR	10:0	0x0	<No Description>

(reserved)	31:11		

CRC_CMDFIFO_ADDR: <No Description>

CRC_CMDFIFO_DOUT MMR:0x0744 MMR_1:0x0744 IND:0x0744 [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CRC_CMDFIFO_DOUT	31:0	0x0	<No Description>

CRC_CMDFIFO_DOUT: <No Description>

PM4_VC_DEBUG_CONFIG MMR:0x07A4 MMR_1:0x07A4 IND:0x07A4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PM4_VC_DONT_START	0	0x0	when set, vertex walker never starts up. 0=Kickoff Vertex Controller upon loading of PM4_VC_CNTL 1=Block Vertex Controller from starting up
PM4_VC_NO_OUTPUT	1	0x0	when set, all vertex walker outputs are dumped. 0=Write output to CMDFIFO 1=Drop all output (i.e. free flow at output port)
PM4_VC_BYPASS_TS_LOGIC	2	0x0	0=Allows timestamp writes only when there is no vertex fetches pending. 1=Allows timestamp writes only when vertex walker is idle. 0=Enable timestamp-specific logic 1=Bypass timestamp-specific logic

PM4_VC_BYPASS_SCP_OPT	3	0x0	0=Write SetupCntlPm4 only when there is a change. (default) 1=Always write SetupCntlPm4 before writing Twicearea. (rage128 behaviour) 0=Send SETUP_CNTL_PM4 only when it's changed 1=Always send SETUP_CNTL_PM4 with TWICEAREA
PM4_VC_FIFO_ARB_MODE	5:4	0x0	0=Automatically, dynamically switch between round-robin and fixed arbitration scheme for optimal performance. 1=Always use round-robin. 2=Always use fixed-priority with vertex walker has higher priority. note: default is '0:auto' while rage128's behaviour is '2:fixed' 0=Automatically, dynamically switch between RR and Fixed schemes for optimal performance 1=Use round-robin scheme to arbitrate data requests between PM4 and Vertex Walker 2=Vertex Walker always has higher priority over PM4 in accessing the shared FIFO
(reserved)	7:6		
PM4_VC_FETCH_DATAFLOW	10:8	0x0	<No Description> 0=Allow BM vertex read request every cycle 1=Allow BM vertex read request every two cycles 2=Allow BM vertex read request every four cycles 3=Allow BM vertex read request every eight cycles 4=Allow BM vertex read request only when all previous requested data have returned 5=Allow BM vertex read request only when all previous requested data have returned and consumed (vertex FIFO empty) 7=Disable budgeting on BM read request. (allow vertex FIFO overflow)
(reserved)	11		
PM4_VC_VBUF_DATAFLOW	14:12	0x0	<No Description> 0=Allow read from Vertex FIFO every cycle 1=Allow read from Vertex FIFO every two cycles 2=Allow read from Vertex FIFO every four cycles 3=Allow read from Vertex FIFO every eight cycles

(reserved)	19:15		
PM4_VC_STAT_MUX	23:20	0x0	<No Description>
PM4_VC_COUNT_SEL	27:24	0x0	<No Description>
(reserved)	30:28		
PM4_VC_COUNT_RESET (W)	31	0x0	This field is write-only and is read back as '0'. This bit resets vertex walker's internal counter. This bit is self-clearing.

PM4_VC_DEBUG_CONFIG: This register is NOT writable by promo4 parser (i.e. PIO only) and readable through 'non-gui' space.
This register is reset with SOFT_RESET_GUI
(all fields default to zero).

PM4_VC_STAT MMR:0x07A8 MMR_1:0x07A8 IND:0x07A8 [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PM4_VC_STAT	31:0	0x0	Reads back some vertex walker's internal status based on the setting of PM4_VC_STAT_MUX@PM4_VC_DEBUG_CONFIG.

PM4_VC_STAT: This register is readonly and readable through 'non-gui' space.

PM4_VC_TIMESTAMP0 MMR:0x07B0 MMR_1:0x07B0 IND:0x07B0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PM4_VC_TIMESTAMP0	31:0	0x0	

PM4_VC_TIMESTAMP0: This register is only writable by promo4 parser and readable through 'non-gui' space.
By default, write to this register is blocked until all previous vertex fetches are returned. If PM4_VC_BYPASS_TS_LOGIC@PM4_VC_DEBUG_CONFIG is set, write to this register is blocked until vertex walker is idle.

PM4_VC_TIMESTAMP1 MMR:0x07B4 MMR_1:0x07B4 IND:0x07B4 [RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
PM4_VC_TIMESTAMP1	31:0	0x0	<No Description>

PM4_VC_TIMESTAMP1: This register is only writable by promo4 parser and readable through 'non-gui' space. By default, write to this register is blocked until all previous vertex fetches are returned. If PM4_VC_BYPASS_TS_LOGIC@PM4_VC_DEBUG_CONFIG is set, write to this register is blocked until vertex walker is idle.

VID_BUFFER_CONTROL MMR:0x0900 MMR_1:0x0900 IND:0x0900 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
CAP0_BUFFER_WATER_MARK	4:0	0x1	<No Description>
(reserved)	7:5		
CAP1_BUFFER_WATER_MARK	12:8	0x1	<No Description>
(reserved)	15:13		
FULL_BUFFER_EN	16	0x0	<No Description> 0=DISABLE 1=ENABLE
(reserved)	19:17		
VID_BUFFER_RESET	20	0x0	<No Description> 0=NOT RESET 1=RESET
(reserved)	23:21		
CAP0_BUFFER_EMPTY (R)	24	0x0	<No Description> 0=EMPTY 1=NOT EMPTY
(reserved)	27:25		
CAP1_BUFFER_EMPTY (R)	28	0x0	<No Description> 0=EMPTY 1=NOT EMPTY
(reserved)	31:29		

VID_BUFFER_CONTROL: <No Description>

BM_QUEUE_FREE_STATUS MMR:0x0A14 MMR_1:0x0A14 IND:0x0A14 [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
BM_VIP0_FREE	3:0	0x2	<No Description>
BM_VIP1_FREE	7:4	0x2	<No Description>
BM_VIP2_FREE	11:8	0x2	<No Description>
BM_VIP3_FREE	15:12	0x2	<No Description>
(reserved)	19:16		
BM_VIDCAP_FREE	23:20	0x8	<No Description>
BM_VIP0_ACTIVE	24	0x0	<No Description> 0=All VIP0 queue transfers are all done 1=A VIP0 queue transfer is active
BM_VIP1_ACTIVE	25	0x0	<No Description> 0=All VIP1 queue transfers are all done 1=A VIP1 queue transfer is active
BM_VIP2_ACTIVE	26	0x0	<No Description> 0=All VIP2 queue transfers are all done 1=A VIP2 queue transfer is active
BM_VIP3_ACTIVE	27	0x0	<No Description> 0=All VIP3 queue transfers are all done 1=A VIP3 queue transfer is active
(reserved)	29:28		
BM_GUI_ACTIVE	30	0x0	<No Description> 0=All GUI queue transfers are all done 1=A GUI queue transfer is active

BM_VIDCAP_ACTIVE	31	0x0	<No Description> 0=All video capture queue transfers are all done 1=A video capture queue transfer is active

BM_QUEUE_FREE_STATUS: <No Description>

BM_ABORT MMR:0x0A88 MMR_1:0x0A88 IND:0x0A88 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
BM_ABORT_QUE	2:0	0x0	<No Description> 0=queue number to abort
(reserved)	3		
BM_ABORT_EN	4	0x0	<No Description> 0=Normal 1=Enable queue abort
(reserved)	31:5		

BM_ABORT: <No Description>

SURFACE_DELAY MMR:0x0B00 MMR_1:0x0B00 IND:0x0B00 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF_POW2_DELAY	3:0	0x1	<No Description>
SURF_NONPOW2_DELAY	7:4	0x3	<No Description>
SURF_TRANSLATION_DIS	8	0x1	<No Description> 0=Enable 1=Disable
(reserved)	31:9		

SURFACE_DELAY: <No Description>

SURFACE0_LOWER_BOUND MMR:0x0B04 MMR_1:0x0B04 IND:0x0B04 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF0_LOWER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE0_LOWER_BOUND: <No Description>

SURFACE1_LOWER_BOUND MMR:0x0B14 MMR_1:0x0B14 IND:0x0B14 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF1_LOWER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE1_LOWER_BOUND: <No Description>

SURFACE2_LOWER_BOUND MMR:0x0B24 MMR_1:0x0B24 IND:0x0B24 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF2_LOWER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE2_LOWER_BOUND: <No Description>

SURFACE3_LOWER_BOUND MMR:0x0B34 MMR_1:0x0B34 IND:0x0B34 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF3_LOWER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO

(reserved)	31:26		

SURFACE3_LOWER_BOUND: <No Description>

SURFACE0_UPPER_BOUND MMR:0x0B08 MMR_1:0x0B08 IND:0x0B08 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF0_UPPER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE0_UPPER_BOUND: <No Description>

SURFACE1_UPPER_BOUND MMR:0x0B18 MMR_1:0x0B18 IND:0x0B18 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF1_UPPER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE1_UPPER_BOUND: <No Description>

SURFACE2_UPPER_BOUND MMR:0x0B28 MMR_1:0x0B28 IND:0x0B28 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF2_UPPER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE2_UPPER_BOUND: <No Description>

SURFACE3_UPPER_BOUND MMR:0x0B38 MMR_1:0x0B38 IND:0x0B38 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

SURF3_UPPER	25:0	0x0	<No Description> NOTE: Bits 5:0 of this field are hardwired to ZERO
(reserved)	31:26		

SURFACE3_UPPER_BOUND: <No Description>

SURFACE0_INFO MMR:0x0B0C MMR_1:0x0B0C IND:0x0B0C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF0_PITCHSEL	4:0	0x0	<No Description> 0=Linear/No translation 1=64 bytes 2=128 bytes 3=256 bytes 4=512 bytes 5=1024 bytes 6=2048 bytes 7=4096 bytes 8=640 bytes 9=1280 bytes 10=2560 bytes 11=5120 bytes 12=1600 bytes 13=3200 bytes 14=6400 bytes 15=832 bytes 16=1664 bytes 17=3328 bytes 18=1920 bytes 19=3840 bytes
(reserved)	31:5		

SURFACE0_INFO: <No Description>

SURFACE1_INFO MMR:0x0B1C MMR_1:0x0B1C IND:0x0B1C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF1_PITCHSEL	4:0	0x0	<No Description> 0=Linear/No translation 1=64 bytes 2=128 bytes 3=256 bytes 4=512 bytes 5=1024 bytes 6=2048 bytes 7=4096 bytes 8=640 bytes 9=1280 bytes 10=2560 bytes 11=5120 bytes 12=1600 bytes 13=3200 bytes 14=6400 bytes 15=832 bytes 16=1664 bytes 17=3328 bytes 18=1920 bytes 19=3840 bytes
(reserved)	31:5		

SURFACE1_INFO: <No Description>

SURFACE2_INFO MMR:0x0B2C MMR_1:0x0B2C IND:0x0B2C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF2_PITCHSEL	4:0	0x0	<No Description> 0=Linear/No translation 1=64 bytes 2=128 bytes 3=256 bytes 4=512 bytes 5=1024 bytes 6=2048 bytes 7=4096 bytes 8=640 bytes 9=1280 bytes 10=2560 bytes 11=5120 bytes 12=1600 bytes 13=3200 bytes 14=6400 bytes 15=832 bytes 16=1664 bytes 17=3328 bytes 18=1920 bytes 19=3840 bytes
(reserved)	31:5		

SURFACE2_INFO: <No Description>

SURFACE3_INFO MMR:0x0B3C MMR_1:0x0B3C IND:0x0B3C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SURF3_PITCHSEL	4:0	0x0	<No Description> 0=Linear/No translation 1=64 bytes 2=128 bytes 3=256 bytes 4=512 bytes 5=1024 bytes 6=2048 bytes 7=4096 bytes 8=640 bytes 9=1280 bytes 10=2560 bytes 11=5120 bytes 12=1600 bytes 13=3200 bytes 14=6400 bytes 15=832 bytes 16=1664 bytes 17=3328 bytes 18=1920 bytes 19=3840 bytes
(reserved)	31:5		

SURFACE3_INFO: <No Description>

AGP_CNTL_B MMR:0x0B44 MMR_1:0x0B44 IND:0x0B44 [RW] 32 bits (access: 8/16/32)			
Field Name	**Bits**	**Default**	**Description**
AGP_0_MISC	0	0x0	<No Description> 0=Normal SBA operation 1=Force generation of full high-mid-low address triplets for all AGP SBA requests.

AGP_1_MISC	1	0x0	<No Description> 0=AGP SBA Bus Enabled 1=AGP SBA Bus Disabled
AGP_2_MISC	2	0x0	<No Description> 0=AGP Revsion ID=2.0 1=AGP Revsion ID=1.0
AGP_3_MISC	3	0x0	<No Description>
AGP_4_MISC	4	0x0	<No Description> 0=Support default AGP transfer rates 1=Disable AGP4X
AGP_5_MISC	5	0x0	<No Description> 0=Support default AGP transfer rates 1=Disable AGP2X and AGP4X
AGP_6_MISC	6	0x0	<No Description> 0=VREF source selected automatically 1=Force AGP receivers to use internal VREF
AGP_7_MISC	7	0x0	<No Description> 0=Normal 1=Set AGP Read Fifo PENDING SLOTS value(0)
AGP_8_MISC	8	0x0	<No Description> 0=Normal 1=Set AGP Read Fifo PENDING SLOTS value(1)
AGP_9_MISC	9	0x0	<No Description> 0=Disable 1=Enable programming PENDING SLOTS values

AGP_A_MISC	10	0x0	<No Description> 0=Normal 1=Extend falling edge of AGP 2X AD strobe output enables
AGP_B_MISC	11	0x0	<No Description> 0=Enable RBFb next-next-next qwsize logic 1=Disable RBFb next-next-next qwsize logic
AGP_C_MISC	12	0x0	<No Description> 0=Normal 1=Enable periodic reset of agp reader DW counters
AGP_D_MISC	13	0x0	<No Description> 0=RBF ignored in BUSCALM calculation 1=Include RBF in BUSCALM calculation.
AGP_E_MISC	14	0x0	<No Description> 0=Enable AGP 4X write bug fix 1=Disable AGP 4X write bug fix
AGP_F_MISC	15	0x0	<No Description> 0=Double flop resync rdptr in AGP req FIFO 1=Single flop resync rdptr in AGP req FIFO
EN_2X_STBB	16	0x0	<No Description> 0=Disable 1=Enable differential AGP strobes in 2X if bus is 1.5V
FORCE_EXT_VREF	17	0x0	<No Description> 0=VREF source selected automatically 1=Force AGP receivers to use external VREF
(reserved)	31:18		

AGP_CNTL_B: <No Description>

FLUSH_1 MMR:0x1704 MMR_1:0x1704 IND:0x1704 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_1	31:0	0x0	Block FIFO'd writes until level 1 engines are idle

FLUSH_1: <No Description>

FLUSH_2 MMR:0x1708 MMR_1:0x1708 IND:0x1708 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_2	31:0	0x0	Block FIFO'd writes until level 2 engines are idle

FLUSH_2: <No Description>

FLUSH_3 MMR:0x170C MMR_1:0x170C IND:0x170C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_3	31:0	0x0	Block FIFO'd writes until level 3 engines are idle

FLUSH_3: <No Description>

FLUSH_4 MMR:0x1710 MMR_1:0x1710 IND:0x1710 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_4	31:0	0x0	Block FIFO'd writes until level 4 engines are idle

FLUSH_4: <No Description>

FLUSH_5 MMR:0x1714 MMR_1:0x1714 IND:0x1714 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_5	31:0	0x0	Block FIFO'd writes until level 5 engines are idle

FLUSH_5: <No Description>

FLUSH_6 MMR:0x1718 MMR_1:0x1718 IND:0x1718 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_6	31:0	0x0	Block FIFO'd writes until level 6 engines are idle

FLUSH_6: <No Description>

FLUSH_7 MMR:0x171C MMR_1:0x171C IND:0x171C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
FLUSH_7	31:0	0x0	Block FIFO'd writes until level 7 engines are idle

FLUSH_7: <No Description>

DST_WIDTH_BW MMR:0x15B4 MMR_1:0x15B4 IND:0x15B4 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_WIDTH	13:0	0x0	Destination width
(reserved)	31:14		

DST_WIDTH_BW: A write to this register indicates all alignment conditions (x, width, scissors, ...) have been met to do a block write fill. It is valid for all memory types, but it is of most value (e.g., Z-clears) in non-byte-maskable memories where, block writes are disabled except when writes to this register occur.

Note: this is an initiator register

LEAD_BRES_LNTH MMR:0x161C MMR_1:0x161C IND:0x161C [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
LEAD_BRES_LNTH	13:0	0x0	Trapezoid leading edge length.
(reserved)	31:14		

LEAD_BRES_LNTH: <No Description>

LEAD_BRES_ERR MMR:0x1600 MMR_1:0x1600 IND:0x1600 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
LEAD_BRES_ERR	19:0	0x0	Bresenham error term for trapezoid leading edge.
(reserved)	31:20		

LEAD_BRES_ERR: <No Description>

LEAD_BRES_INC MMR:0x1604 MMR_1:0x1604 IND:0x1604 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
LEAD_BRES_INC	19:0	0x0	Bresenham increment for trapezoid leading edge.
(reserved)	31:20		

LEAD_BRES_INC: <No Description>

LEAD_BRES_DEC MMR:0x1608 MMR_1:0x1608 IND:0x1608 [W] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
LEAD_BRES_DEC	19:0	0x0	Bresenham decrement for trapezoid leading edge.
(reserved)	31:20		

LEAD_BRES_DEC: <No Description>

TRAIL_X MMR:0x1618 MMR_1:0x1618 IND:0x1618 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
TRAIL_X	13:0	0x0	X for trapezoid trailing edge.
(reserved)	31:14		

TRAIL_X: <No Description>

TRAIL_BRES_ERR MMR:0x160C MMR_1:0x160C IND:0x160C [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
TRAIL_BRES_ERR	19:0	0x0	Bresenham error term for trapezoid trailing edge.
(reserved)	31:20		

TRAIL_BRES_ERR: <No Description>

TRAIL_BRES_INC MMR:0x1610 MMR_1:0x1610 IND:0x1610 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
TRAIL_BRES_INC	19:0	0x0	Bresenham increment for line and Trapezoid trailing edge.
(reserved)	31:20		

TRAIL_BRES_INC: <No Description>

TRAIL_BRES_DEC MMR:0x1614 MMR_1:0x1614 IND:0x1614 [RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
TRAIL_BRES_DEC	19:0	0x0	Bresenham decrement for line and Trapezoid trailing edge.
(reserved)	31:20		

TRAIL_BRES_DEC: <No Description>

TRAIL_X_SUB MMR:0x1620 MMR_1:0x1620 IND:0x1620
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
TRAIL_X_FRACT	3:0	0x0	Sub pixel bits of TRAIL_X coordinate. Note that when TRAIL_X is written these bits are set to 1000 (one half)
TRAIL_X	17:4	0x0	Trailing edge X coordinate: range -8192 to 8191. Aliased to TRAIL_X[13:0]
(reserved)	31:18		

TRAIL_X_SUB: <No Description>

LEAD_BRES_LNTH_SUB MMR:0x1624 MMR_1:0x1624 IND:0x1624
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
LEAD_BRES_LNTH_SUB	3:0	0x0	Trapezoid leading edge length.
LEAD_BRES_LNTH	17:4	0x0	Trapezoid leading edge length. Aliased to DST_BRES_LNTH[13:0]
(reserved)	31:18		

LEAD_BRES_LNTH_SUB: <No Description>

WAIT_UNTIL MMR:0x1720 MMR_1:0x1720 IND:0x1720
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description

EVENT_CRTC_OFFSET	0	0x0	Used to stall until the display has started displaying the last new CRTC_OFFSET value written. (i.e. used to do page flips.) Write 0: No effect. Write 1: Stall cmdfifo until CRTC_GUI_TRIG_OFFSET = 0. See CRTC_OFFSET register.
EVENT_RE_CRTC_VLINE	1	0x0	Used to stall until the display has reached the start of a specific range of raster lines. Write 0: No effect. Write 1: Stall cmdfifo until CRTC_GUI_TRIG_VLINE has a rising edge. See CRTC_GUI_TRIG_VLINE register.
EVENT_FE_CRTC_VLINE	2	0x0	Used to stall until the display has reached the end of a specific range of raster lines. Write 0: No effect. Write 1: Stall cmdfifo until CRTC_GUI_TRIG_VLINE has a falling edge. See CRTC_GUI_TRIG_VLINE register.
EVENT_CRTC_VLINE	3	0x0	Used to stall until the display has reached anywhere in a specific range of raster lines. Write 0: No effect. Write 1: Stall cmdfifo until CRTC_GUI_TRIG_VLINE = 1. See CRTC_GUI_TRIG_VLINE register.
EVENT_BM_VIP0_IDLE	4	0x0	Write 0: No effect Write 1: Stall cmdfifo 'til BM_IDLE for this channel.
EVENT_BM_VIP1_IDLE	5	0x0	Write 0: No effect Write 1: Stall cmdfifo 'til BM_IDLE for this channel.
EVENT_BM_VIP2_IDLE	6	0x0	Write 0: No effect Write 1: Stall cmdfifo 'til BM_IDLE for this channel.
EVENT_BM_VIP3_IDLE	7	0x0	Write 0: No effect Write 1: Stall cmdfifo 'til BM_IDLE for this channel.

EVENT_BM_VIDCAP_IDLE	8	0x0	Write 0: No effect Write 1: Stall cmdfifo 'til BM_IDLE for this channel.
EVENT_BM_GUI_IDLE	9	0x0	Write 0: No effect Write 1: Stall 'til BM_IDLE for this channel.
EVENT_CMDFIFO	10	0x0	Write 0: No effect Write 1: Stall cmdfifo 'til number of entries specified in EVENT_CMDFIFO_ENTRIES is met.
EVENT_OV0_FLIP	11	0x0	Write 0: No effect. Write 1: Stall cmdfifo 'til OV0_FLIP='1' 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_FLIP' 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_FLIP will go low at unlock and then high during VBlank (when the hardware double buffering flips the registers). The behavior of OV0_FLIP is undefined if OV0_STALL_GUI_UNTIL_FLIP is written to when the lock bit is not set. OV0_FLIP 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).
(reserved)	19:12		
EVENT_CMDFIFO_ENTRIES	26:20	0x0	Number of cmdfifo entries to trigger on.
(reserved)	31:27		

WAIT_UNTIL: Enables stalling the processing of commands out of the command FIFO until the selected trigger condition is reached. This is done to delay the processing of further contents of the PM4 stream until certain engines in the chip have reached certain milestones. Stall cmdfifo based on 'AND' of all set triggers.

GUI_SCRATCH_REG0 MMR:0x15E0 MMR_1:0x15E0 IND:0x15E0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

GUI_SCRATCH_REG0	31:0	0x0	FIFO'd scratch register

GUI_SCRATCH_REG0: <No Description>

GUI_SCRATCH_REG1 MMR:0x15E4 MMR_1:0x15E4 IND:0x15E4
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
GUI_SCRATCH_REG1	31:0	0x0	FIFO'd scratch register

GUI_SCRATCH_REG1: <No Description>

GUI_SCRATCH_REG2 MMR:0x15E8 MMR_1:0x15E8 IND:0x15E8
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
GUI_SCRATCH_REG2	31:0	0x0	FIFO'd scratch register

GUI_SCRATCH_REG2: <No Description>

GUI_SCRATCH_REG3 MMR:0x15EC MMR_1:0x15EC IND:0x15EC
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
GUI_SCRATCH_REG3	31:0	0x0	FIFO'd scratch register

GUI_SCRATCH_REG3: <No Description>

GUI_SCRATCH_REG4 MMR:0x15F0 MMR_1:0x15F0 IND:0x15F0
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
GUI_SCRATCH_REG4	31:0	0x0	FIFO'd scratch register

GUI_SCRATCH_REG4: <No Description>

GUI_SCRATCH_REG5 MMR:0x15F4 MMR_1:0x15F4 IND:0x15F4
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
GUI_SCRATCH_REG5	31:0	0x0	FIFO'd scratch register

GUI_SCRATCH_REG5: <No Description>

DEFAULT_OFFSET MMR:0x16E0 MMR_1:0x16E0 IND:0x16E0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_OFFSET	25:0	0x0	Default destination offset address for DP_GUI_MASTER_CNTL operations. See description of DST_OFFSET. NOTE: Bits 3:0 of this field are hardwired to ZERO
(reserved)	31:26		

DEFAULT_OFFSET: <No Description>

DEFAULT_PITCH MMR:0x16E4 MMR_1:0x16E4 IND:0x16E4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_PITCH	9:0	0x0	Default destination pitch for DP_GUI_MASTER_CNTL operations. See description of DST_PITCH.
(reserved)	15:10		
DEFAULT_TILE	16	0x0	<No Description>
(reserved)	31:17		

DEFAULT_PITCH: <No Description>

DEFAULT_SC_BOTTOM_RIGHT MMR:0x16E8 MMR_1:0x16E8 IND:0x16E8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DEFAULT_SC_RIGHT	13:0	0x0	Default right scissor for DP_GUI_MASTER_CNTL
(reserved)	15:14		
DEFAULT_SC_BOTTOM	29:16	0x0	Default bottom scissor for DP_GUI_MASTER_CNTL
(reserved)	31:30		

DEFAULT_SC_BOTTOM_RIGHT: <No Description>

GUI_STAT MMR:0x1740 MMR_1:0x1740 IND:0x1740 [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
GUI_FIFOCNT	11:0	0x40	Number of free CMDFIFO entries
(reserved)	15:12		
PM4_BUSY	16	0x0	State of PROMO_4 engine
MICRO_BUSY	17	0x0	State of the micro engine
FPU_BUSY	18	0x0	State of the pre-setup engine
VC_BUSY	19	0x0	State of the Vertex controller engine
IDCT_BUSY	20	0x0	State of the IDCT engine
ENG_EV_BUSY	21	0x0	State of the event engine
SETUP_BUSY	22	0x0	State of the setup engine
EDGEWALK_BUSY	23	0x0	State of the edgewalker pipeline
ADDRESSING_BUSY	24	0x0	State of the texel/Destination addressing pipeline
ENG_3D_BUSY	25	0x0	State of the eng_3d data pipeline
ENG_2D_SM_BUSY	26	0x0	State of the eng_2d engine
ENG_2D_BUSY	27	0x0	State of the eng_2d pipeline
GUI_WB_BUSY	28	0x0	State of the gui write buffer
CACHE_BUSY	29	0x0	State of the pixel cache
(reserved)	30		
GUI_ACTIVE	31	0x0	'OR' of the above bits

GUI_STAT: <No Description>

GUI_DEBUG0 MMR:0x16A0 MMR_1:0x16A0 IND:0x16A0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
TRIPLICATION_DIS	0	0x0	disable monochrome HOST_DATA triplication for 24BPP destinations
BLOCK_WRITE_DIS	1	0x0	disable automatic SGRAM block writes
COLOR_WRITE_DIS	2	0x0	disable SGRAM color register write operations
BRUSH_TRANSPARENT_DIS	3	0x0	<No Description>
HOST_ZWS_FIX_DIS	4	0x0	<No Description>
HOST_EARLY_FIX_DIS	5	0x0	<No Description>
GUI_DEBUG6	6	0x0	<No Description>
BLKWRT_ZWS_FIX_DIS	7	0x0	<No Description>
GUI_DEBUG8	8	0x0	<No Description>
CLR_CMP_CLIP_FIX_DIS	9	0x0	<No Description>
GUI_DEBUG10	10	0x0	<No Description>
XCHECK_FIX_DIS	11	0x0	<No Description>
GUI_DEBUG12	12	0x0	<No Description>
GUI_DEBUG13	13	0x0	<No Description>
GUI_DEBUG14	14	0x0	<No Description>
GUI_DEBUG15	15	0x0	<No Description>
GUI_DEBUG16	16	0x0	<No Description>
GUI_DEBUG17	17	0x0	<No Description>
GUI_DEBUG18	18	0x0	<No Description>

GUI_DEBUG19	19	0x0	<No Description>
GUI_DEBUG20	20	0x0	<No Description>
GUI_DEBUG21	21	0x0	<No Description>
GUI_DEBUG22	22	0x0	<No Description>
GUI_DEBUG23	23	0x0	<No Description>
GUI_DEBUG24	24	0x0	<No Description>
GUI_DEBUG25	25	0x0	<No Description>
GUI_DEBUG26	26	0x0	<No Description>
GUI_DEBUG27	27	0x0	<No Description>
GUI_DEBUG28	28	0x0	<No Description>
GUI_DEBUG29	29	0x0	<No Description>
GUI_DEBUG30	30	0x0	<No Description>
GUI_DEBUG31	31	0x0	<No Description>

GUI_DEBUG0: 2D engine debug bits. These bits can only be written when the GUI is idle and are written through the command FIFO.

GUI_DEBUG1 MMR:0x16A4 MMR_1:0x16A4 IND:0x16A4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

GUI_DEBUG1: Reserved for future use

GUI_DEBUG2 MMR:0x16A8 MMR_1:0x16A8 IND:0x16A8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

GUI_DEBUG2: Reserved for future use

GUI_DEBUG3 MMR:0x16AC MMR_1:0x16AC IND:0x16AC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

GUI_DEBUG3: Reserved for future use

GUI_DEBUG4 MMR:0x16B0 MMR_1:0x16B0 IND:0x16B0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

GUI_DEBUG4: Reserved for future use

GUI_DEBUG5 MMR:0x16B4 MMR_1:0x16B4 IND:0x16B4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

GUI_DEBUG5: Reserved for future use

GUI_DEBUG6 MMR:0x16B8 MMR_1:0x16B8 IND:0x16B8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

GUI_DEBUG6: Reserved for future use

GUI_PROBE MMR:0x16BC MMR_1:0x16BC IND:0x16BC [R] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

GUI_STATE	2:0	0x0	eng_2d state machine 000 = idle 001 = DO_COLOR_WRITE 010 = DO_SPAN_EVEN 011 = DO_SPAN 100-101 = (undefined) 110 = WAIT_FOR_SPAN 111 = WAIT_FOR_PIPE_EMPTY
GUI_PROBE_DUMMY3	3	0x0	<No Description>
GUI_PROBE_DUMMY4	4	0x0	<No Description>
GUI_PROBE_DUMMY5	5	0x0	<No Description>
GUI_PROBE_DUMMY6	6	0x0	<No Description>
GUI_PROBE_DUMMY7	7	0x0	<No Description>
GUI_SPAN_REQ	8	0x0	Span FIFO request
GUI_SPAN_RDY	9	0x0	Span FIFO ready
GUI_REQ_SRCS	10	0x0	all required sources present (stage 1 write)
GUI_HOST_REQ	11	0x0	HOST_DATA request
GUI_HOST_RDY	12	0x0	HOST_DATA ready
GUI_SRC_REQ	13	0x0	SRC/Z request
GUI_SRC_RDY	14	0x0	SRC/Z ready
GUI_E3D_REQ	15	0x0	3D data request
GUI_E3D_RDY	16	0x0	3D data ready
GUI_DST_REQ	17	0x0	DST request
GUI_DST_RDY	18	0x0	DST ready
GUI_WRT_REQ	19	0x0	DST write request

GUI_WRT_ZS_REQ	20	0x0	Z write request
GUI_WRT_RDY	21	0x0	Write ready
GUI_PROBE_DUMMY22	22	0x0	<No Description>
GUI_PROBE_DUMMY23	23	0x0	<No Description>
GUI_PROBE_DUMMY24	24	0x0	<No Description>
GUI_PROBE_DUMMY25	25	0x0	<No Description>
GUI_PROBE_DUMMY26	26	0x0	<No Description>
GUI_PROBE_DUMMY27	27	0x0	<No Description>
GUI_PROBE_DUMMY28	28	0x0	<No Description>
GUI_PROBE_DUMMY29	29	0x0	<No Description>
GUI_PROBE_DUMMY30	30	0x0	<No Description>
GUI_PROBE_DUMMY31	31	0x0	<No Description>

GUI_PROBE: probe of internal 2D draw engine signals. not for public use!

WINDOW_XY_OFFSET MMR:0x1BCC MMR_1:0x1BCC IND:0x1BCC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
WINDOW_OFFSET_Y	15:0	0x0	Destination Y coordinate offset from Window Origin in S.12.2 format or S.11.4 format depending on SUB_PIX_AMNT.
WINDOW_OFFSET_X	31:16	0x0	Destination X coordinate offset from Window Origin in S.12.2 format or S.11.4 format depending on SUB_PIX_AMNT.

WINDOW_XY_OFFSET: This register specifies factors added to all X, Y vertex values to align them to the window.

DRAW_LINE_POINT MMR:0x1BD0 MMR_1:0x1BD0 IND:0x1BD0 [RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
TYPE_SELECT	0	0x0	<No Description> 0=Draw Point 1=Draw Line
STARTING_VERTEX_SELECT	2:1	0x0	<No Description> 1=Start at Vertex 1 2=Start at Vertex 2 3=Start at Vertex 3
ENDING_VERTEX_SELECT	4:3	0x0	<No Description> 1=End at Vertex 1 2=End at Vertex 2 3=End at Vertex 3
DRAW_LAST_PEL	5	0x0	<No Description> 0=Don't draw last pixel 1=Draw last pixel
FLAT_SHADE_FN_LINE	7:6	0x0	<No Description> 0=Flat Shading Off 1=Use vertex 1 color/alpha 2=Use vertex 2 color/alpha 3=Use vertex 3 color/alpha
SOLID_MODE_LINE	8	0x0	<No Description> 0=Solid Mode Off 1=Solid Mode On
(reserved)	15:9		
LINE_SLOPE	31:16	0x0	<No Description>

DRAW_LINE_POINT: <No Description>

W_START MMR:0x18CC MMR_1:0x18CC IND:0x18CC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**

W_START	27:0	0x0	Initial value of W coordinate address. In S.0.M27 'pseudo floating point format'.
(reserved)	31:28		

W_START: <No Description>

CONSTANT_COLOR MMR:0x1A30 MMR_1:0x1A30 IND:0x1A30 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
CONSTANT_BLUE	7:0	0x0	Constant color field used in texture combining operations.
CONSTANT_GREEN	15:8	0x0	Constant color field used in texture combining operations.
CONSTANT_RED	23:16	0x0	Constant color field used in texture combining operations.
CONSTANT_ALPHA	31:24	0x0	Constant color field used in texture combining and lighting operations.

CONSTANT_COLOR: <No Description>

Z_VIS MMR:0x1AD4 MMR_1:0x1AD4 IND:0x1AD4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
Z_VIS	0	0x0	<No Description>
(reserved)	31:1		

Z_VIS: <No Description>

PC_GUI_MODE MMR:0x1744 MMR_1:0x1744 IND:0x1744 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PC_GUI_PRIORITY	0	0x0	<No Description>
PC_RISE_DF_EN	1	0x0	<No Description>
PC_FALL_DF_EN	2	0x0	<No Description>

PC_BYPASS_EN	3	0x0	<No Description>
PC_CACHE_SIZE	4	0x0	<No Description>
PC_IGNORE_UNIFY	5	0x0	<No Description>
PC_IGNORE_WRHINT	6	0x0	<No Description>
PC_IGNORE_RDHINT	7	0x0	<No Description>
PC_RISE_DP_EN	8	0x0	<No Description>
PC_7P2_MODE	9	0x0	<No Description>
PC_DEBUG_EN	10	0x0	<No Description>
PC_DEBUG_SEL	14:11	0x0	<No Description>
(reserved)	31:15		

PC_GUI_MODE: <No Description>

PC_NGUI_MODE MMR:0x0180 MMR_1:0x0180 IND:0x0180 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PC_GUI_PRIORITY	0	0x0	<No Description>
PC_RISE_DF_EN	1	0x0	<No Description>
PC_FALL_DF_EN	2	0x0	<No Description>
PC_BYPASS_EN	3	0x0	<No Description>
PC_CACHE_SIZE	4	0x0	<No Description>
PC_IGNORE_UNIFY	5	0x0	<No Description>
PC_IGNORE_WRHINT	6	0x0	<No Description>
PC_IGNORE_RDHINT	7	0x0	<No Description>
PC_RISE_DP_EN	8	0x0	<No Description>

PC_7P2_MODE	9	0x0	<No Description>
(reserved)	31:10		

PC_NGUI_MODE: <No Description>

PC_GUI_CTLSTAT MMR:0x1748 MMR_1:0x1748 IND:0x1748 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PC_FLUSH_GUI	1:0	0x0	<No Description>
PC_RI_GUI	3:2	0x0	<No Description>
PC_FLUSH_NONGUI	5:4	0x0	<No Description>
PC_RI_NONGUI	7:6	0x0	<No Description>
PC_PURGE_GUI	8	0x0	<No Description>
PC_PURGE_NONGUI	9	0x0	<No Description>
(reserved)	23:10		
PC_DIRTY	24	0x0	<No Description>
PC_PURGE_DOPURGE	25	0x0	<No Description>
PC_FLUSH_DOFLUSH	26	0x0	<No Description>
PC_BUSY_INIT (R)	27	0x0	<No Description>
PC_BUSY_FLUSH (R)	28	0x0	<No Description>
PC_BUSY_GUI (R)	29	0x0	<No Description>
PC_BUSY_NGUI (R)	30	0x0	<No Description>
PC_BUSY (R)	31	0x0	<No Description>

PC_GUI_CTLSTAT: <No Description>

PC_NGUI_CTLSTAT MMR:0x0184 MMR_1:0x0184 IND:0x0184 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PC_FLUSH_GUI	1:0	0x0	<No Description>
PC_RI_GUI	3:2	0x0	<No Description>
PC_FLUSH_NONGUI	5:4	0x0	<No Description>
PC_RI_NONGUI	7:6	0x0	<No Description>
PC_PURGE_GUI	8	0x0	<No Description>
PC_PURGE_NONGUI	9	0x0	<No Description>
(reserved)	23:10		
PC_DIRTY	24	0x0	<No Description>
PC_PURGE_DOPURGE	25	0x0	<No Description>
PC_FLUSH_DOFLUSH	26	0x0	<No Description>
PC_BUSY_INIT (R)	27	0x0	<No Description>
PC_BUSY_FLUSH (R)	28	0x0	<No Description>
PC_BUSY_GUI (R)	29	0x0	<No Description>
PC_BUSY_NGUI (R)	30	0x0	<No Description>
PC_BUSY (R)	31	0x0	<No Description>

PC_NGUI_CTLSTAT: <No Description>

PC_MISC_CTL MMR:0x0188 MMR_1:0x0188 IND:0x0188 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
PC_OFF0_UP_SEL	4:0	0x0	<No Description>
(reserved)	5		

PC_LRU_FIX_DIS	6	0x0	<No Description>
PC_MODESWSTALL_DIS	7	0x0	<No Description>
PC_OFF0_DN_SEL	12:8	0x0	<No Description>
PC_XLAT_STALL_FIX_DIS	13	0x0	<No Description>
PC_LOW32K_CHECK_DIS	14	0x0	<No Description>
(reserved)	15		
PC_OFF1_UP_SEL	20:16	0x0	<No Description>
(reserved)	23:21		
PC_OFF1_DN_SEL	28:24	0x0	<No Description>
(reserved)	31:29		

PC_MISC_CTL: <No Description>

PC_DEBUG_MODE MMR:0x1760 MMR_1:0x1760 IND:0x1760 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
(reserved)	31:0		

PC_DEBUG_MODE: <No Description>

MISC_3D_STATE_CNTL_REG MMR:0x1CA0 MMR_1:0x1CA0 IND:0x1CA0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
REF_ALPHA	7:0	0x0	Alpha reference value used when alpha compare enabled.
SCALE_3D_FN	9:8	0x0	The SCALE_3D_FCN encodes the operation(s) to be performed by the 3D / Scaling pipe. 0 = No operation 1 = Scaling 2 = Texture Mapping/Shading 3 = (Reserved) Note that if this field is set to 0, many 3D/Front-End Scalar/Setup Engine registers are NOT writeable. Hence this field should be written to a non-zero value prior to trying to write any other 3D/Front-End Scalar registers. This field is set to 0 on Chip Reset. 0=No operation 1=Scaling 2=Texture Mapping/Shading
SCALE_PIX_REP	10	0x0	During Scaling operations, replicate pixels rather than linear blend. 0=Blend pixels during scale 1=Replicate pixels during scale
(reserved)	11		

ALPHA_COMB_FCN	13:12	0x0	Allows modification of how the ALPHA_BLND_SRC and ALPHA_BLND_DST are combined: 0 = Add and Clamp 1 = Add but no Clamp 2 = Subtract Dst from Src, and clamp 3 = Subtract Dst from Src but don’t clamp 0=Add and Clamp 1=Add but no Clamp 2=Subtract Dst from Src and clamp 3=Subtract Dst from Src but don't clamp
FOG_TABLE_EN	14	0x0	0 = FOG_VERTEX 1 = FOG_TABLE 0=Use Vertex Fog 1=Use Fog Table based on Z interpolator value
(reserved)	15		

ALPHA_BLND_SRC	19:16	0x0	Determines the type of SRC alpha blending to use: 0 = BLEND_ZERO 1 = BLEND_ONE 2 = BLEND_SRCCOLOR 3 = BLEND_INVSRCCOLOR 4 = BLEND_SRCALPHA 5 = BLEND_INVSRCALPHA 6 = BLEND_DESTALPHA 7 = BLEND_INVDESTALPHA 8 = BLEND_DESTCOLOR 9 = BLEND_INVDESTCOLOR 0a = BLEND_SRCALPHASAT 0b = BLRND_BOTHSRCALPHA 0c = BLEND_BOTHINVSRCALPHA 0d-0f = Reserved 0=Blend factor is (0,0,0,0) 1=Blend factor is (1,1,1,1) 2=Blend factor is (RS,GS,BS,AS) 3=Blend factor is (1-RS,1-GS,1-BS,1-AS) 4=Blend factor is (AS, AS, AS, AS) 5=Blend factor is (1-AS,1-AS,1-As,1-AS) 6=Blend factor is (Ad, Ad, Ad, Ad) 7=Blend factor is (1- Ad,1- Ad,1- Ad,1- Ad) 8=Blend factor is (Rd,Gd,Bd,Ad) 9=Blend factor is (1-Rd,1-Gd,1-Bd,1-Ad) 10=Blend factor is (f,f,f,1);f=min(AS, 1-Ad) 11=SRC Blend factor is (AS,AS,AS,AS), force DST Blend factor to (1-AS,1-AS,1-AS, 1-AS) 12=SRC Blend factor is (1-AS,1-AS,1-AS,1-AS), force DST Blend factor to (AS,AS,AS, AS)

ALPHA_BLND_DST	23:20	0x0	Determines the type of DEST alpha blending to use: 0 = BLEND_ZERO 1 = BLEND_ONE 2 = BLEND_SRCCOLOR 3 = BLEND_INVSRCCOLOR 4 = BLEND_SRCALPHA 5 = BLEND_INVSRCALPHA 6 = BLEND_DESTALPHA 7 = BLEND_INVDESTALPHA 8 = BLEND_DESTCOLOR 9 = BLEND_INVDESTCOLOR 0a = BLEND_SRCALPHASAT 0b-0f = Reserved 0=Blend factor is (0,0,0,0) 1=Blend factor is (1,1,1,1) 2=Blend factor is (RS,GS,BS,AS) 3=Blend factor is (1-RS,1-GS,1-BS,1-AS) 4=Blend factor is (AS, AS, AS, AS) 5=Blend factor is (1-AS,1-AS,1- AS) 6=Blend factor is (Ad, Ad, Ad, Ad) 7=Blend factor is (1- Ad,1- Ad,1- Ad,1- Ad) 8=Blend factor is (Rd,Gd,Bd,Ad) 9=Blend factor is (1-Rd,1-Gd,1-Bd,1-Ad) 10=Blend factor is (f,f,f,1);f=min(AS, 1-Ad)
ALPHA_TEST_OP	26:24	0x0	Specifies what function to use when comparing the SRC Alpha value against a specified Alpha value: 0=Never Pass 1=Src < Ref 2=Src <= Ref 3=Src == Ref 4=Src >= Ref 5=Src > Ref 6=Src != Ref 7=Always Pass
(reserved)	29:27		

CLR_CMP_FCN_3D	31:30	0x0	NOTE: This type of color keying is unavailable when using the old texture interface (execute buffer, DrawPrimitv etc). When the new multi-texture API is used, the APP must use the texel alpha. This is what MS is advocating. Aliased to CLR_CMP_CNTL_3D) bits 1:0 0 = False 1 = True 2 =Texel != CLR_CMP_CLR_3D 3 = Texel = CLR_CMP_CLR_3D 0=False 1=True 2=Texel != CLR_CMP_CLR_3D 3=Texel = CLR_CMP_CLR_3D

MISC_3D_STATE_CNTL_REG: <No Description>

CONSTANT_COLOR_C MMR:0x1D34 MMR_1:0x1D34 IND:0x1D34
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
CONSTANT_BLUE	7:0	0x0	Blue component of constant color that can be used by texture combining.
CONSTANT_GREEN	15:8	0x0	Green component of constant color that can be used by texture combining.
CONSTANT_RED	23:16	0x0	Red component of constant color that can be used by texture combining.
CONSTANT_ALPHA	31:24	0x0	Alpha component of constant color that can be used by texture combining.

CONSTANT_COLOR_C: <No Description>

PLANE_3D_MASK_C MMR:0x1D44 MMR_1:0x1D44 IND:0x1D44
[RW] 32 bits (access: 32)

Field Name	Bits	Default	Description
PLANE_3D_MASK	31:0	0x0	This MASK is used to perform bitmask operations on color planes. I.E. This value would be written into DP_WRITE_MASK.

PLANE_3D_MASK_C: <No Description>

MC_SRC1_CNTL MMR:0x19D8 MMR_1:0x19D8 IND:0x19D8 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SCALE_HACC	12:0	0x0	<No Description>
(reserved)	15:13		
SCALE_VACC	27:16	0x0	<No Description>
IDCT_EN	28	0x0	<No Description>
SECONDARY_TEX_EN	29	0x0	<No Description> 0=No secondary texture 1=Use secondary texture
SCALE_PITCH_ADJ	31:30	0x0	<No Description>

MC_SRC1_CNTL: <No Description>

MC_SRC2_CNTL MMR:0x19D4 MMR_1:0x19D4 IND:0x19D4 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SECONDARY_SCALE_HACC	12:0	0x0	<No Description>
(reserved)	15:13		
SECONDARY_SCALE_VACC	27:16	0x0	<No Description>
(reserved)	29:28		
SECONDARY_SCALE_PITCH_ADJ	31:30	0x0	<No Description>

MC_SRC2_CNTL: <No Description>

MC_DST_CNTL MMR:0x19DC MMR_1:0x19DC IND:0x19DC [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
DST_Y	13:0	0x0	<No Description>
(reserved)	15:14		

DST_X	29:16	0x0	<No Description>
DST_PITCH_ADJ	31:30	0x0	<No Description>

MC_DST_CNTL: <No Description>

MC_START_CNTL MMR:0x19E0 MMR_1:0x19E0 IND:0x19E0 [RW] 32 bits (access: 32)			
Field Name	**Bits**	**Default**	**Description**
SCALE_OFFSET_PTR	3:0	0x0	<No Description>
DST_OFFSET	24:4	0x0	<No Description>
ALPHA_EN	25	0x0	<No Description> 0=Alpha blending off 1=Alpha blending on
SECONDARY_OFFSET_PTR	28:26	0x0	<No Description>
DST_HEIGHT_WIDTH	31:29	0x0	<No Description>

MC_START_CNTL: <No Description>

Appendix A
Cross Referenced Index

A.1 All Registers Sorted by Name

Table A-1 All Registers Sorted by Name

Register Name
ADAPTER_ID
ADAPTER_ID_W
AGP_APER_OFFSET
AGP_BASE
AGP_CNTL
AGP_CNTL_B
AGP_COMMAND
AGP_PLL_CNTL
AGP_STATUS
AMCGPIO_A_MIR
AMCGPIO_A_REG
AMCGPIO_EN_MIR
AMCGPIO_EN_REG
AMCGPIO_MASK
AMCGPIO_MASK_MIR
AMCGPIO_Y_MIR
AMCGPIO_Y_REG
ATTR00
ATTR10
ATTR11
ATTR12
ATTR13
ATTR14
ATTRDR
ATTRDW
ATTRX
AUX_SC_CNTL
AUX1_SC_BOTTOM
AUX1_SC_LEFT
AUX1_SC_RIGHT
AUX1_SC_TOP
AUX2_SC_BOTTOM
AUX2_SC_LEFT
AUX2_SC_RIGHT
AUX2_SC_TOP

Table A-1 All Registers Sorted by Name (Continued)

Register Name
AUX3_SC_BOTTOM
AUX3_SC_LEFT
AUX3_SC_RIGHT
AUX3_SC_TOP
BASE_CODE
BIOS_0_SCRATCH
BIOS_1_SCRATCH
BIOS_2_SCRATCH
BIOS_3_SCRATCH
BIOS_ROM
BIST
BM_ABORT
BM_QUEUE_FREE_STATUS
BUS_CNTL
BUS_CNTL1
CACHE_LINE
CAPABILITIES_ID
CAPABILITIES_PTR
CLK_PIN_CNTL
CLOCK_CNTL_DATA
CLOCK_CNTL_INDEX
CLR_CMP_CLR_3D
CLR_CMP_CLR_DST
CLR_CMP_CLR_SRC
CLR_CMP_CNTL
CLR_CMP_MSK
CLR_CMP_MSK_3D
COMMAND
COMPOSITE_SHADOW_ID
CONFIG_APER_0_BASE
CONFIG_APER_1_BASE
CONFIG_APER_SIZE
CONFIG_BONDS
CONFIG_CNTL
CONFIG_MEMSIZE
CONFIG_MEMSIZE_EMBEDDED
CONFIG_REG_1_BASE
CONFIG_REG_APER_SIZE
CONFIG_XSTRAP
CONSTANT_COLOR
CONSTANT_COLOR_C
CRC_CMDFIFO_ADDR

Table A-1 All Registers Sorted by Name (Continued)

Register Name
CRC_CMDFIFO_DOUT
CRT00
CRT00_S
CRT01
CRT01_S
CRT02
CRT02_S
CRT03
CRT03_S
CRT04
CRT04_S
CRT05
CRT05_S
CRT06
CRT06_S
CRT07
CRT07_S
CRT08
CRT08_S
CRT09
CRT09_S
CRT0A
CRT0A_S
CRT0B
CRT0B_S
CRT0C
CRT0C_S
CRT0D
CRT0D_S
CRT0E
CRT0E_S
CRT0F
CRT0F_S
CRT10
CRT10_S
CRT11
CRT11_S
CRT12
CRT12_S
CRT13
CRT13_S

Table A-1 All Registers Sorted by Name (Continued)

Register Name
CRT14
CRT14_S
CRT15
CRT15_S
CRT16
CRT16_S
CRT17
CRT17_S
CRT18
CRT18_S
CRT1E
CRT1E_S
CRT1F
CRT1F_S
CRT22
CRT22_S
CRTC_CRNT_FRAME
CRTC_DEBUG
CRTC_EXT_CNTL
CRTC_GEN_CNTL
CRTC_GUI_TRIG_VLINE
CRTC_H_SYNC_STRT_WID
CRTC_H_TOTAL_DISP
CRTC_OFFSET
CRTC_OFFSET_CNTL
CRTC_PITCH
CRTC_STATUS
CRTC_V_SYNC_STRT_WID
CRTC_V_TOTAL_DISP
CRTC_VLINE_CRNT_VLINE
CRTC8_DATA
CRTC8_IDX
CUR_CLR0
CUR_CLR1
CUR_HORZ_VERT_OFF
CUR_HORZ_VERT_POSN
CUR_OFFSET
DAC_CNTL
DAC_CRC_SIG
DAC_DATA
DAC_EXT_CNTL
DAC_MASK

Table A-1 All Registers Sorted by Name (Continued)

Register Name
DAC_R_INDEX
DAC_W_INDEX
DDA_CONFIG
DDA_ON_OFF
DEFAULT_OFFSET
DEFAULT_PITCH
DEFAULT_SC_BOTTOM_RIGHT
DEVICE_ID
DP_BRUSH_BKGD_CLR
DP_BRUSH_FRGD_CLR
DP_CNTL
DP_CNTL_XDIR_YDIR_YMAJOR
DP_DATATYPE
DP_GUI_MASTER_CNTL
DP_GUI_MASTER_CNTL_C
DP_MIX
DP_SRC_BKGD_CLR
DP_SRC_FRGD_CLR
DP_WRITE_MSK
DRAW_LINE_POINT
DST_BRES_DEC
DST_BRES_ERR
DST_BRES_INC
DST_BRES_LNTH
DST_BRES_LNTH_SUB
DST_HEIGHT
DST_HEIGHT_WIDTH
DST_HEIGHT_WIDTH_8
DST_HEIGHT_Y
DST_OFFSET
DST_PITCH
DST_PITCH_OFFSET
DST_PITCH_OFFSET_C
DST_WIDTH
DST_WIDTH_BW
DST_WIDTH_HEIGHT
DST_WIDTH_X
DST_WIDTH_X_INCY
DST_X
DST_X_SUB
DST_X_Y

Table A-1 All Registers Sorted by Name (Continued)

Register Name
DST_Y
DST_Y_SUB
DST_Y_X
EXT_MEM_CNTL
FCP_CNTL
FLUSH_1
FLUSH_2
FLUSH_3
FLUSH_4
FLUSH_5
FLUSH_6
FLUSH_7
GEN_INT_CNTL
GEN_INT_STATUS
GEN_RESET_CNTL
GEN_STATUS
GENENB
GENFC_RD
GENFC_WT
GENMO_RD
GENMO_WT
GENS0
GENS1
GPIO_MONID
GRA00
GRA01
GRA02
GRA03
GRA04
GRA05
GRA06
GRA07
GRA08
GRPH8_DATA
GRPH8_IDX
GUI_DEBUG0
GUI_DEBUG1
GUI_DEBUG2
GUI_DEBUG3
GUI_DEBUG4
GUI_DEBUG5
GUI_DEBUG6

Table A-1 All Registers Sorted by Name (Continued)

Register Name
GUI_PROBE
GUI_SCRATCH_REG0
GUI_SCRATCH_REG1
GUI_SCRATCH_REG2
GUI_SCRATCH_REG3
GUI_SCRATCH_REG4
GUI_SCRATCH_REG5
GUI_STAT
HEADER
HOST_DATA_LAST
HOST_DATA0
HOST_DATA1
HOST_DATA2
HOST_DATA3
HOST_DATA4
HOST_DATA5
HOST_DATA6
HOST_DATA7
HOST_PATH_CNTL
HTOTAL_CNTL
HW_DEBUG
HW_DEBUG2
INTERRUPT_LINE
INTERRUPT_PIN
IO_BASE
LATENCY
LEAD_BRES_DEC
LEAD_BRES_ERR
LEAD_BRES_INC
LEAD_BRES_LNTH
LEAD_BRES_LNTH_SUB
MAX_LATENCY
MC_DST_CNTL
MC_SRC1_CNTL
MC_SRC2_CNTL
MC_START_CNTL
MCLK_CNTL
MDGPIO_A_REG
MDGPIO_EN_REG
MDGPIO_MASK
MDGPIO_Y_REG

Table A-1 All Registers Sorted by Name (Continued)

Register Name
MEM_ADDR_CONFIG
MEM_BASE
MEM_CNTL
MEM_INIT_LAT_TIMER
MEM_INTF_CNTL
MEM_SDRAM_MODE_REG
MEM_STR_CNTL
MEM_VGA_RP_SEL
MEM_VGA_WP_SEL
MIN_GRANT
MISC_3D_STATE_CNTL_REG
MM_DATA
MM_INDEX
MPLL_CNTL
N_VIF_COUNT
OV0_COL_CONV
OVR_CLR
OVR_WID_LEFT_RIGHT
OVR_WID_TOP_BOTTOM
PAD_AGPINPUT_DELAY
PAD_CTLR_STRENGTH
PAD_CTLR_UPDATE
PALETTE_DATA
PALETTE_INDEX
PC_DEBUG_MODE
PC_GUI_CTLSTAT
PC_GUI_MODE
PC_MISC_CTL
PC_NGUI_CTLSTAT
PC_NGUI_MODE
PCI_GART_PAGE
PLANE_3D_MASK_C
PLL_TEST_CNTL
PM4_BUFFER_DL_WPTR_DELAY
PM4_VC_DEBUG_CONFIG
PM4_VC_STAT
PM4_VC_TIMESTAMP0
PM4_VC_TIMESTAMP1
PMI_CAP_ID
PMI_DATA
PMI_NXT_CAP_PTR
PMI_PMC_REG

Table A-1 All Registers Sorted by Name (Continued)

Register Name
PMI_PMCSR_REG
PPLL_CNTL
PPLL_DIV_0
PPLL_DIV_1
PPLL_DIV_2
PPLL_DIV_3
PPLL_REF_DIV
REG_BASE
REGPROG_INF
REVISION_ID
SC_BOTTOM
SC_BOTTOM_RIGHT
SC_BOTTOM_RIGHT_C
SC_LEFT
SC_RIGHT
SC_TOP
SC_TOP_LEFT
SC_TOP_LEFT_C
SECONDARY_SCALE_HACC
SECONDARY_SCALE_PITCH
SECONDARY_SCALE_VACC
SECONDARY_SCALE_X_INC
SECONDARY_SCALE_Y_INC
SEPROM_CNTL
SEQ00
SEQ01
SEQ02
SEQ03
SEQ04
SEQ8_DATA
SEQ8_IDX
SNAPSHOT_F_COUNT
SNAPSHOT_VH_COUNTS
SNAPSHOT_VIF_COUNT
SRC_OFFSET
SRC_PITCH
SRC_PITCH_OFFSET
SRC_SC_BOTTOM
SRC_SC_BOTTOM_RIGHT
SRC_SC_RIGHT
SRC_X

Table A-1 All Registers Sorted by Name (Continued)

Register Name
SRC_X_Y
SRC_Y
SRC_Y_X
STATUS
SUB_CLASS
SURFACE_DELAY
SURFACE0_INFO
SURFACE0_LOWER_BOUND
SURFACE0_UPPER_BOUND
SURFACE1_INFO
SURFACE1_LOWER_BOUND
SURFACE1_UPPER_BOUND
SURFACE2_INFO
SURFACE2_LOWER_BOUND
SURFACE2_UPPER_BOUND
SURFACE3_INFO
SURFACE3_LOWER_BOUND
SURFACE3_UPPER_BOUND
SW_SEMAPHORE
TEST_DEBUG_CNTL
TEST_DEBUG_MUX
TEST_DEBUG_OUT
TRAIL_BRES_DEC
TRAIL_BRES_ERR
TRAIL_BRES_INC
TRAIL_X
TRAIL_X_SUB
VCLK_ECP_CNTL
VENDOR_ID
VGA_DDA_CONFIG
VGA_DDA_ON_OFF
VID_BUFFER_CONTROL
VIDEOMUX_CNTL
VIPH_CH0_DATA
VIPH_CH1_DATA
VIPH_CH2_DATA
VIPH_CH3_DATA
W_START
WAIT_UNTIL
WINDOW_XY_OFFSET
X_MPLL_REF_FB_DIV
XCLK_CNTL

Table A-1 All Registers Sorted by Name (Continued)

Register Name
XDLL_CNTL
XPLL_CNTL
Z_VIS

A.2 MMR Registers Sorted by Name

Table A-2 MMR Registers Sorted by Name

Register Name	Address	Secondary Adress	Indexed Address	Page
AGP_APER_OFFSET	MMR:0x0178		IND:0x0178	3-187
AGP_BASE	MMR:0x0170		IND:0x0170	3-186
AGP_CNTL	MMR:0x0174		IND:0x0174	3-186
AGP_CNTL_B	MMR:0x0B44		IND:0x0B44	3-231
AMCGPIO_A_MIR	MMR:0x00A0	IOR:0x00A0	IND:0x00A0	3-198
AMCGPIO_A_REG	MMR:0x01A0		IND:0x01A0	3-211
AMCGPIO_EN_MIR	MMR:0x00A8	IOR:0x00A8	IND:0x00A8	3-199
AMCGPIO_EN_REG	MMR:0x01A8		IND:0x01A8	3-212
AMCGPIO_MASK	MMR:0x0194		IND:0x0194	3-209
AMCGPIO_MASK_MIR	MMR:0x009C	IOR:0x009C	IND:0x009C	3-198
AMCGPIO_Y_MIR	MMR:0x00A4	IOR:0x00A4	IND:0x00A4	3-199
AMCGPIO_Y_REG	MMR:0x01A4		IND:0x01A4	3-211
AUX_SC_CNTL	MMR:0x1660		IND:0x1660	3-156
AUX1_SC_BOTTOM	MMR:0x1670		IND:0x1670	3-158
AUX1_SC_LEFT	MMR:0x1664		IND:0x1664	3-157
AUX1_SC_RIGHT	MMR:0x1668		IND:0x1668	3-157
AUX1_SC_TOP	MMR:0x166C		IND:0x166C	3-158
AUX2_SC_BOTTOM	MMR:0x1680		IND:0x1680	3-159
AUX2_SC_LEFT	MMR:0x1674		IND:0x1674	3-158
AUX2_SC_RIGHT	MMR:0x1678		IND:0x1678	3-158
AUX2_SC_TOP	MMR:0x167C		IND:0x167C	3-158
AUX3_SC_BOTTOM	MMR:0x1690		IND:0x1690	3-160
AUX3_SC_LEFT	MMR:0x1684		IND:0x1684	3-159
AUX3_SC_RIGHT	MMR:0x1688		IND:0x1688	3-159
AUX3_SC_TOP	MMR:0x168C		IND:0x168C	3-159
BIOS_0_SCRATCH	MMR:0x0010	IOR:0x0010	IND:0x0010	3-85
BIOS_1_SCRATCH	MMR:0x0014	IOR:0x0014	IND:0x0014	3-85
BIOS_2_SCRATCH	MMR:0x0018	IOR:0x0018	IND:0x0018	3-85
BIOS_3_SCRATCH	MMR:0x001C	IOR:0x001C	IND:0x001C	3-85
BM_ABORT	MMR:0x0A88		IND:0x0A88	3-224
BM_QUEUE_FREE_STATUS	MMR:0x0A14		IND:0x0A14	3-223
BUS_CNTL	MMR:0x0030	IOR:0x0030		3-106
BUS_CNTL1	MMR:0x0034	IOR:0x0034		3-191

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
CLOCK_CNTL_DATA	MMR:0x000C	IOR:0x000C	IND:0x000C	3-86
CLOCK_CNTL_INDEX	MMR:0x0008	IOR:0x0008	IND:0x0008	3-86
CLR_CMP_CLR_3D	MMR:0x1A24		IND:0x1A24	3-179
CLR_CMP_CLR_DST	MMR:0x15C8		IND:0x15C8	3-178
CLR_CMP_CLR_SRC	MMR:0x15C4		IND:0x15C4	3-178
CLR_CMP_CNTL	MMR:0x15C0		IND:0x15C0	3-178
CLR_CMP_MSK	MMR:0x15CC		IND:0x15CC	3-179
CLR_CMP_MSK_3D	MMR:0x1A28		IND:0x1A28	3-180
COMPOSITE_SHADOW_ID	MMR:0x1A0C		IND:0x1A0C	3-145
CONFIG_APER_0_BASE	MMR:0x0100		IND:0x0100	3-12
CONFIG_APER_1_BASE	MMR:0x0104		IND:0x0104	3-12
CONFIG_APER_SIZE	MMR:0x0108		IND:0x0108	3-202
CONFIG_BONDS	MMR:0x00E8	IOR:0x00E8	IND:0x00E8	3-11
CONFIG_CNTL	MMR:0x00E0	IOR:0x00E0		3-10
CONFIG_MEMSIZE	MMR:0x00F8	IOR:0x00F8	IND:0x00F8	3-12
CONFIG_MEMSIZE_EMBEDDED	MMR:0x0114		IND:0x0114	3-203
CONFIG_REG_1_BASE	MMR:0x010C		IND:0x010C	3-202
CONFIG_REG_APER_SIZE	MMR:0x0110		IND:0x0110	3-202
CONFIG_XSTRAP	MMR:0x00E4	IOR:0x00E4	IND:0x00E4	3-11
CONSTANT_COLOR	MMR:0x1A30		IND:0x1A30	3-251
CONSTANT_COLOR_C	MMR:0x1D34		IND:0x1D34	3-260
CRC_CMDFIFO_ADDR	MMR:0x0740		IND:0x0740	3-218
CRC_CMDFIFO_DOUT	MMR:0x0744		IND:0x0744	3-219
CRTC_CRNT_FRAME	MMR:0x0214		IND:0x0214	3-214
CRTC_DEBUG	MMR:0x021C		IND:0x021C	3-49
CRTC_EXT_CNTL	MMR:0x0054	IOR:0x0054	IND:0x0054	3-63
CRTC_GEN_CNTL	MMR:0x0050	IOR:0x0050		3-61
CRTC_GUI_TRIG_VLINE	MMR:0x0218		IND:0x0218	3-69
CRTC_H_SYNC_STRT_WID	MMR:0x0204		IND:0x0204	3-67
CRTC_H_TOTAL_DISP	MMR:0x0200		IND:0x0200	3-66
CRTC_OFFSET	MMR:0x0224		IND:0x0224	3-70
CRTC_OFFSET_CNTL	MMR:0x0228		IND:0x0228	3-71
CRTC_PITCH	MMR:0x022C		IND:0x022C	3-74
CRTC_STATUS	MMR:0x005C	IOR:0x005C	IND:0x005C	3-66
CRTC_V_SYNC_STRT_WID	MMR:0x020C		IND:0x020C	3-68
CRTC_V_TOTAL_DISP	MMR:0x0208		IND:0x0208	3-67
CRTC_VLINE_CRNT_VLINE	MMR:0x0210		IND:0x0210	3-68
CUR_CLR0	MMR:0x026C		IND:0x026C	3-83
CUR_CLR1	MMR:0x0270		IND:0x0270	3-83
CUR_HORZ_VERT_OFF	MMR:0x0268		IND:0x0268	3-82
CUR_HORZ_VERT_POSN	MMR:0x0264		IND:0x0264	3-81
CUR_OFFSET	MMR:0x0260		IND:0x0260	3-81

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
DAC_CNTL	MMR:0x0058	IOR:0x0058		3-123
DAC_CRC_SIG	MMR:0x02CC		IND:0x02CC	3-126
DAC_EXT_CNTL	MMR:0x0280		IND:0x0280	3-217
DDA_CONFIG	MMR:0x02E0		IND:0x02E0	3-76
DDA_ON_OFF	MMR:0x02E4		IND:0x02E4	3-76
DEFAULT_OFFSET	MMR:0x16E0		IND:0x16E0	3-243
DEFAULT_PITCH	MMR:0x16E4		IND:0x16E4	3-243
DEFAULT_SC_BOTTOM_RIGHT	MMR:0x16E8		IND:0x16E8	3-243
DP_BRUSH_BKGD_CLR	MMR:0x1478		IND:0x1478	3-164
DP_BRUSH_FRGD_CLR	MMR:0x147C		IND:0x147C	3-164
DP_CNTL	MMR:0x16C0		IND:0x16C0	3-165
DP_CNTL_XDIR_YDIR_YMAJOR	MMR:0x16D0		IND:0x16D0	3-170
DP_DATATYPE	MMR:0x16C4		IND:0x16C4	3-168
DP_GUI_MASTER_CNTL	MMR:0x146C		IND:0x146C	3-171
DP_GUI_MASTER_CNTL_C	MMR:0x1C84		IND:0x1C84	3-175
DP_MIX	MMR:0x16C8		IND:0x16C8	3-171
DP_SRC_BKGD_CLR	MMR:0x15DC		IND:0x15DC	3-164
DP_SRC_FRGD_CLR	MMR:0x15D8		IND:0x15D8	3-164
DP_WRITE_MSK	MMR:0x16CC		IND:0x16CC	3-171
DRAW_LINE_POINT	MMR:0x1BD0		IND:0x1BD0	3-249
DST_BRES_DEC	MMR:0x1630		IND:0x1630	3-142
DST_BRES_ERR	MMR:0x1628		IND:0x1628	3-142
DST_BRES_INC	MMR:0x162C		IND:0x162C	3-142
DST_BRES_LNTH	MMR:0x1634		IND:0x1634	3-142
DST_BRES_LNTH_SUB	MMR:0x1638		IND:0x1638	3-144
DST_HEIGHT	MMR:0x1410		IND:0x1410	3-139
DST_HEIGHT_WIDTH	MMR:0x143C		IND:0x143C	3-140
DST_HEIGHT_WIDTH_8	MMR:0x158C		IND:0x158C	3-140
DST_HEIGHT_Y	MMR:0x15A0		IND:0x15A0	3-141
DST_OFFSET	MMR:0x1404		IND:0x1404	3-137
DST_PITCH	MMR:0x1408		IND:0x1408	3-137
DST_PITCH_OFFSET	MMR:0x142C		IND:0x142C	3-137
DST_PITCH_OFFSET_C	MMR:0x1C80		IND:0x1C80	3-145
DST_WIDTH	MMR:0x140C		IND:0x140C	3-139
DST_WIDTH_BW	MMR:0x15B4		IND:0x15B4	3-236
DST_WIDTH_HEIGHT	MMR:0x1598		IND:0x1598	3-140
DST_WIDTH_X	MMR:0x1588		IND:0x1588	3-141
DST_WIDTH_X_INCY	MMR:0x159C		IND:0x159C	3-141
DST_X	MMR:0x141C		IND:0x141C	3-138
DST_X_SUB	MMR:0x15A4		IND:0x15A4	3-144
DST_X_Y	MMR:0x1594		IND:0x1594	3-138

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
DST_Y	MMR:0x1420		IND:0x1420	3-138
DST_Y_SUB	MMR:0x15A8		IND:0x15A8	3-144
DST_Y_X	MMR:0x1438		IND:0x1438	3-139
EXT_MEM_CNTL	MMR:0x0144		IND:0x0144	3-113
FLUSH_1	MMR:0x1704		IND:0x1704	3-234
FLUSH_2	MMR:0x1708		IND:0x1708	3-234
FLUSH_3	MMR:0x170C		IND:0x170C	3-234
FLUSH_4	MMR:0x1710		IND:0x1710	3-234
FLUSH_5	MMR:0x1714		IND:0x1714	3-234
FLUSH_6	MMR:0x1718		IND:0x1718	3-235
FLUSH_7	MMR:0x171C		IND:0x171C	3-235
GEN_INT_CNTL	MMR:0x0040	IOR:0x0040	IND:0x0040	3-192
GEN_INT_STATUS	MMR:0x0044	IOR:0x0044	IND:0x0044	3-193
GEN_RESET_CNTL	MMR:0x00F0	IOR:0x00F0	IND:0x00F0	3-200
GEN_STATUS	MMR:0x00F4	IOR:0x00F4	IND:0x00F4	3-201
GPIO_MONID	MMR:0x0068	IOR:0x0068	IND:0x0068	3-196
GUI_DEBUG0	MMR:0x16A0		IND:0x16A0	3-245
GUI_DEBUG1	MMR:0x16A4		IND:0x16A4	3-246
GUI_DEBUG2	MMR:0x16A8		IND:0x16A8	3-246
GUI_DEBUG3	MMR:0x16AC		IND:0x16AC	3-247
GUI_DEBUG4	MMR:0x16B0		IND:0x16B0	3-247
GUI_DEBUG5	MMR:0x16B4		IND:0x16B4	3-247
GUI_DEBUG6	MMR:0x16B8		IND:0x16B8	3-247
GUI_PROBE	MMR:0x16BC		IND:0x16BC	3-247
GUI_SCRATCH_REG0	MMR:0x15E0		IND:0x15E0	3-240
GUI_SCRATCH_REG1	MMR:0x15E4		IND:0x15E4	3-241
GUI_SCRATCH_REG2	MMR:0x15E8		IND:0x15E8	3-241
GUI_SCRATCH_REG3	MMR:0x15EC		IND:0x15EC	3-241
GUI_SCRATCH_REG4	MMR:0x15F0		IND:0x15F0	3-241
GUI_SCRATCH_REG5	MMR:0x15F4		IND:0x15F4	3-241
GUI_STAT	MMR:0x1740		IND:0x1740	3-244
HOST_DATA_LAST	MMR:0x17E0		IND:0x17E0	3-153
HOST_DATA0	MMR:0x17C0		IND:0x17C0	3-151
HOST_DATA1	MMR:0x17C4		IND:0x17C4	3-151
HOST_DATA2	MMR:0x17C8		IND:0x17C8	3-151
HOST_DATA3	MMR:0x17CC		IND:0x17CC	3-152
HOST_DATA4	MMR:0x17D0		IND:0x17D0	3-152
HOST_DATA5	MMR:0x17D4		IND:0x17D4	3-153
HOST_DATA6	MMR:0x17D8		IND:0x17D8	3-153
HOST_DATA7	MMR:0x17DC		IND:0x17DC	3-153
HOST_PATH_CNTL	MMR:0x0130		IND:0x0130	3-149
HW_DEBUG	MMR:0x0128		IND:0x0128	3-133

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
HW_DEBUG2	MMR:0x011C		IND:0x011C	3-203
LEAD_BRES_DEC	MMR:0x1608		IND:0x1608	3-237
LEAD_BRES_ERR	MMR:0x1600		IND:0x1600	3-236
LEAD_BRES_INC	MMR:0x1604		IND:0x1604	3-236
LEAD_BRES_LNTH	MMR:0x161C		IND:0x161C	3-236
LEAD_BRES_LNTH_SUB	MMR:0x1624		IND:0x1624	3-238
MC_DST_CNTL	MMR:0x19DC		IND:0x19DC	3-261
MC_SRC1_CNTL	MMR:0x19D8		IND:0x19D8	3-261
MC_SRC2_CNTL	MMR:0x19D4		IND:0x19D4	3-261
MC_START_CNTL	MMR:0x19E0		IND:0x19E0	3-262
MDGPIO_A_REG	MMR:0x01AC		IND:0x01AC	3-212
MDGPIO_EN_REG	MMR:0x01B0		IND:0x01B0	3-213
MDGPIO_MASK	MMR:0x0198		IND:0x0198	3-210
MDGPIO_Y_REG	MMR:0x01B4		IND:0x01B4	3-213
MEM_ADDR_CONFIG	MMR:0x0148		IND:0x0148	3-75
MEM_CNTL	MMR:0x0140		IND:0x0140	3-110
MEM_INIT_LAT_TIMER	MMR:0x0154		IND:0x0154	3-122
MEM_INTF_CNTL	MMR:0x014C		IND:0x014C	3-115
MEM_SDRAM_MODE_REG	MMR:0x0158		IND:0x0158	3-205
MEM_STR_CNTL	MMR:0x0150		IND:0x0150	3-120
MEM_VGA_RP_SEL	MMR:0x003C	IOR:0x003C	IND:0x003C	3-110
MEM_VGA_WP_SEL	MMR:0x0038	IOR:0x0038	IND:0x0038	3-110
MISC_3D_STATE_CNTL_REG	MMR:0x1CA0		IND:0x1CA0	3-256
MM_DATA	MMR:0x0004	IOR:0x0004		3-191
MM_INDEX	MMR:0x0000	IOR:0x0000		3-191
N_VIF_COUNT	MMR:0x0248		IND:0x0248	3-216
OV0_COL_CONV	MMR:0x04FC		IND:0x04FC	3-217
OVR_CLR	MMR:0x0230		IND:0x0230	3-79
OVR_WID_LEFT_RIGHT	MMR:0x0234		IND:0x0234	3-79
OVR_WID_TOP_BOTTOM	MMR:0x0238		IND:0x0238	3-79
PAD_AGPINPUT_DELAY	MMR:0x0164		IND:0x0164	3-206
PAD_CTLR_STRENGTH	MMR:0x0168		IND:0x0168	3-206
PAD_CTLR_UPDATE	MMR:0x016C		IND:0x016C	3-207
PALETTE_DATA	MMR:0x00B4	IOR:0x00B4	IND:0x00B4	3-200
PALETTE_INDEX	MMR:0x00B0	IOR:0x00B0	IND:0x00B0	3-199
PC_DEBUG_MODE	MMR:0x1760		IND:0x1760	3-256
PC_GUI_CTLSTAT	MMR:0x1748		IND:0x1748	3-253
PC_GUI_MODE	MMR:0x1744		IND:0x1744	3-251
PC_MISC_CTL	MMR:0x0188		IND:0x0188	3-254
PC_NGUI_CTLSTAT	MMR:0x0184		IND:0x0184	3-254
PC_NGUI_MODE	MMR:0x0180		IND:0x0180	3-252

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
PCI_GART_PAGE	MMR:0x017C		IND:0x017C	3-207
PLANE_3D_MASK_C	MMR:0x1D44		IND:0x1D44	3-260
PM4_BUFFER_DL_WPTR_DELAY	MMR:0x0718		IND:0x0718	3-218
PM4_VC_DEBUG_CONFIG	MMR:0x07A4		IND:0x07A4	3-219
PM4_VC_STAT	MMR:0x07A8		IND:0x07A8	3-221
PM4_VC_TIMESTAMP0	MMR:0x07B0		IND:0x07B0	3-221
PM4_VC_TIMESTAMP1	MMR:0x07B4		IND:0x07B4	3-221
SC_BOTTOM	MMR:0x164C		IND:0x164C	3-155
SC_BOTTOM_RIGHT	MMR:0x16F0		IND:0x16F0	3-162
SC_BOTTOM_RIGHT_C	MMR:0x1C8C		IND:0x1C8C	3-162
SC_LEFT	MMR:0x1640		IND:0x1640	3-155
SC_RIGHT	MMR:0x1644		IND:0x1644	3-155
SC_TOP	MMR:0x1648		IND:0x1648	3-155
SC_TOP_LEFT	MMR:0x16EC		IND:0x16EC	3-161
SC_TOP_LEFT_C	MMR:0x1C88		IND:0x1C88	3-162
SECONDARY_SCALE_HACC	MMR:0x198C		IND:0x198C	3-161
SECONDARY_SCALE_PITCH	MMR:0x1980		IND:0x1980	3-160
SECONDARY_SCALE_VACC	MMR:0x1990		IND:0x1990	3-161
SECONDARY_SCALE_X_INC	MMR:0x1984		IND:0x1984	3-160
SECONDARY_SCALE_Y_INC	MMR:0x1988		IND:0x1988	3-161
SEPROM_CNTL	MMR:0x006C	IOR:0x006C	IND:0x006C	3-197
SNAPSHOT_F_COUNT	MMR:0x0244		IND:0x0244	3-215
SNAPSHOT_VH_COUNTS	MMR:0x0240		IND:0x0240	3-215
SNAPSHOT_VIF_COUNT	MMR:0x024C		IND:0x024C	3-216
SRC_OFFSET	MMR:0x15AC		IND:0x15AC	3-146
SRC_PITCH	MMR:0x15B0		IND:0x15B0	3-146
SRC_PITCH_OFFSET	MMR:0x1428		IND:0x1428	3-146
SRC_SC_BOTTOM	MMR:0x165C		IND:0x165C	3-148
SRC_SC_BOTTOM_RIGHT	MMR:0x16F4		IND:0x16F4	3-148
SRC_SC_RIGHT	MMR:0x1654		IND:0x1654	3-148
SRC_X	MMR:0x1414		IND:0x1414	3-147
SRC_X_Y	MMR:0x1590		IND:0x1590	3-147
SRC_Y	MMR:0x1418		IND:0x1418	3-147
SRC_Y_X	MMR:0x1434		IND:0x1434	3-147
SURFACE_DELAY	MMR:0x0B00		IND:0x0B00	3-224
SURFACE0_INFO	MMR:0x0B0C		IND:0x0B0C	3-228
SURFACE0_LOWER_BOUND	MMR:0x0B04		IND:0x0B04	3-225
SURFACE0_UPPER_BOUND	MMR:0x0B08		IND:0x0B08	3-226
SURFACE1_INFO	MMR:0x0B1C		IND:0x0B1C	3-229
SURFACE1_LOWER_BOUND	MMR:0x0B14		IND:0x0B14	3-225
SURFACE1_UPPER_BOUND	MMR:0x0B18		IND:0x0B18	3-226
SURFACE2_INFO	MMR:0x0B2C		IND:0x0B2C	3-230

Table A-2 MMR Registers Sorted by Name (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
SURFACE2_LOWER_BOUND	MMR:0x0B24		IND:0x0B24	3-225
SURFACE2_UPPER_BOUND	MMR:0x0B28		IND:0x0B28	3-226
SURFACE3_INFO	MMR:0x0B3C		IND:0x0B3C	3-231
SURFACE3_LOWER_BOUND	MMR:0x0B34		IND:0x0B34	3-225
SURFACE3_UPPER_BOUND	MMR:0x0B38		IND:0x0B38	3-226
SW_SEMAPHORE	MMR:0x013C		IND:0x013C	3-205
TEST_DEBUG_CNTL	MMR:0x0120		IND:0x0120	3-129
TEST_DEBUG_MUX	MMR:0x0124		IND:0x0124	3-132
TEST_DEBUG_OUT	MMR:0x012C		IND:0x012C	3-204
TRAIL_BRES_DEC	MMR:0x1614		IND:0x1614	3-237
TRAIL_BRES_ERR	MMR:0x160C		IND:0x160C	3-237
TRAIL_BRES_INC	MMR:0x1610		IND:0x1610	3-237
TRAIL_X	MMR:0x1618		IND:0x1618	3-237
TRAIL_X_SUB	MMR:0x1620		IND:0x1620	3-238
VGA_DDA_CONFIG	MMR:0x02E8		IND:0x02E8	3-77
VGA_DDA_ON_OFF	MMR:0x02EC		IND:0x02EC	3-77
VID_BUFFER_CONTROL	MMR:0x0900		IND:0x0900	3-222
VIDEOMUX_CNTL	MMR:0x0190		IND:0x0190	3-208
VIPH_CH0_DATA	MMR:0x0C00		IND:0x0C00	3-213
VIPH_CH1_DATA	MMR:0x0C04		IND:0x0C04	3-214
VIPH_CH2_DATA	MMR:0x0C08		IND:0x0C08	3-214
VIPH_CH3_DATA	MMR:0x0C0C		IND:0x0C0C	3-214
W_START	MMR:0x18CC		IND:0x18CC	3-250
WAIT_UNTIL	MMR:0x1720		IND:0x1720	3-238
WINDOW_XY_OFFSET	MMR:0x1BCC		IND:0x1BCC	3-249
Z_VIS	MMR:0x1AD4		IND:0x1AD4	3-251

A.3 MMR Registers Sorted by Address

Table A-3 MMR Registers Sorted by Address

Register Name	Address	Secondary Adress	Indexed Address	Page
MM_INDEX	MMR:0x0000	IOR:0x0000		3-191
MM_DATA	MMR:0x0004	IOR:0x0004		3-191
CLOCK_CNTL_INDEX	MMR:0x0008	IOR:0x0008	IND:0x0008	3-86
CLOCK_CNTL_DATA	MMR:0x000C	IOR:0x000C	IND:0x000C	3-86
BIOS_0_SCRATCH	MMR:0x0010	IOR:0x0010	IND:0x0010	3-85
BIOS_1_SCRATCH	MMR:0x0014	IOR:0x0014	IND:0x0014	3-85
BIOS_2_SCRATCH	MMR:0x0018	IOR:0x0018	IND:0x0018	3-85
BIOS_3_SCRATCH	MMR:0x001C	IOR:0x001C	IND:0x001C	3-85
BUS_CNTL	MMR:0x0030	IOR:0x0030		3-106
BUS_CNTL1	MMR:0x0034	IOR:0x0034		3-191
MEM_VGA_WP_SEL	MMR:0x0038	IOR:0x0038	IND:0x0038	3-110
MEM_VGA_RP_SEL	MMR:0x003C	IOR:0x003C	IND:0x003C	3-110
GEN_INT_CNTL	MMR:0x0040	IOR:0x0040	IND:0x0040	3-192
GEN_INT_STATUS	MMR:0x0044	IOR:0x0044	IND:0x0044	3-193
CRTC_GEN_CNTL	MMR:0x0050	IOR:0x0050		3-61
CRTC_EXT_CNTL	MMR:0x0054	IOR:0x0054	IND:0x0054	3-63
DAC_CNTL	MMR:0x0058	IOR:0x0058		3-123
CRTC_STATUS	MMR:0x005C	IOR:0x005C	IND:0x005C	3-66
GPIO_MONID	MMR:0x0068	IOR:0x0068	IND:0x0068	3-196
SEPROM_CNTL	MMR:0x006C	IOR:0x006C	IND:0x006C	3-197
AMCGPIO_MASK_MIR	MMR:0x009C	IOR:0x009C	IND:0x009C	3-198
AMCGPIO_A_MIR	MMR:0x00A0	IOR:0x00A0	IND:0x00A0	3-198
AMCGPIO_Y_MIR	MMR:0x00A4	IOR:0x00A4	IND:0x00A4	3-199
AMCGPIO_EN_MIR	MMR:0x00A8	IOR:0x00A8	IND:0x00A8	3-199
PALETTE_INDEX	MMR:0x00B0	IOR:0x00B0	IND:0x00B0	3-199
PALETTE_DATA	MMR:0x00B4	IOR:0x00B4	IND:0x00B4	3-200
CONFIG_CNTL	MMR:0x00E0	IOR:0x00E0		3-10
CONFIG_XSTRAP	MMR:0x00E4	IOR:0x00E4	IND:0x00E4	3-11
CONFIG_BONDS	MMR:0x00E8	IOR:0x00E8	IND:0x00E8	3-11
GEN_RESET_CNTL	MMR:0x00F0	IOR:0x00F0	IND:0x00F0	3-200
GEN_STATUS	MMR:0x00F4	IOR:0x00F4	IND:0x00F4	3-201
CONFIG_MEMSIZE	MMR:0x00F8	IOR:0x00F8	IND:0x00F8	3-12
CONFIG_APER_0_BASE	MMR:0x0100		IND:0x0100	3-12
CONFIG_APER_1_BASE	MMR:0x0104		IND:0x0104	3-12
CONFIG_APER_SIZE	MMR:0x0108		IND:0x0108	3-202
CONFIG_REG_1_BASE	MMR:0x010C		IND:0x010C	3-202
CONFIG_REG_APER_SIZE	MMR:0x0110		IND:0x0110	3-202
CONFIG_MEMSIZE_EMBEDDED	MMR:0x0114		IND:0x0114	3-203
HW_DEBUG2	MMR:0x011C		IND:0x011C	3-203

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
TEST_DEBUG_CNTL	MMR:0x0120		IND:0x0120	3-129
TEST_DEBUG_MUX	MMR:0x0124		IND:0x0124	3-132
HW_DEBUG	MMR:0x0128		IND:0x0128	3-133
TEST_DEBUG_OUT	MMR:0x012C		IND:0x012C	3-204
HOST_PATH_CNTL	MMR:0x0130		IND:0x0130	3-149
SW_SEMAPHORE	MMR:0x013C		IND:0x013C	3-205
MEM_CNTL	MMR:0x0140		IND:0x0140	3-110
EXT_MEM_CNTL	MMR:0x0144		IND:0x0144	3-113
MEM_ADDR_CONFIG	MMR:0x0148		IND:0x0148	3-75
MEM_INTF_CNTL	MMR:0x014C		IND:0x014C	3-115
MEM_STR_CNTL	MMR:0x0150		IND:0x0150	3-120
MEM_INIT_LAT_TIMER	MMR:0x0154		IND:0x0154	3-122
MEM_SDRAM_MODE_REG	MMR:0x0158		IND:0x0158	3-205
PAD_AGPINPUT_DELAY	MMR:0x0164		IND:0x0164	3-206
PAD_CTLR_STRENGTH	MMR:0x0168		IND:0x0168	3-206
PAD_CTLR_UPDATE	MMR:0x016C		IND:0x016C	3-207
AGP_BASE	MMR:0x0170		IND:0x0170	3-186
AGP_CNTL	MMR:0x0174		IND:0x0174	3-186
AGP_APER_OFFSET	MMR:0x0178		IND:0x0178	3-187
PCI_GART_PAGE	MMR:0x017C		IND:0x017C	3-207
PC_NGUI_MODE	MMR:0x0180		IND:0x0180	3-252
PC_NGUI_CTLSTAT	MMR:0x0184		IND:0x0184	3-254
PC_MISC_CTL	MMR:0x0188		IND:0x0188	3-254
VIDEOMUX_CNTL	MMR:0x0190		IND:0x0190	3-208
AMCGPIO_MASK	MMR:0x0194		IND:0x0194	3-209
MDGPIO_MASK	MMR:0x0198		IND:0x0198	3-210
AMCGPIO_A_REG	MMR:0x01A0		IND:0x01A0	3-211
AMCGPIO_Y_REG	MMR:0x01A4		IND:0x01A4	3-211
AMCGPIO_EN_REG	MMR:0x01A8		IND:0x01A8	3-212
MDGPIO_A_REG	MMR:0x01AC		IND:0x01AC	3-212
MDGPIO_EN_REG	MMR:0x01B0		IND:0x01B0	3-213
MDGPIO_Y_REG	MMR:0x01B4		IND:0x01B4	3-213
CRTC_H_TOTAL_DISP	MMR:0x0200		IND:0x0200	3-66
CRTC_H_SYNC_STRT_WID	MMR:0x0204		IND:0x0204	3-67
CRTC_V_TOTAL_DISP	MMR:0x0208		IND:0x0208	3-67
CRTC_V_SYNC_STRT_WID	MMR:0x020C		IND:0x020C	3-68
CRTC_VLINE_CRNT_VLINE	MMR:0x0210		IND:0x0210	3-68
CRTC_CRNT_FRAME	MMR:0x0214		IND:0x0214	3-214
CRTC_GUI_TRIG_VLINE	MMR:0x0218		IND:0x0218	3-69
CRTC_DEBUG	MMR:0x021C		IND:0x021C	3-49
CRTC_OFFSET	MMR:0x0224		IND:0x0224	3-70

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
CRTC_OFFSET_CNTL	MMR:0x0228		IND:0x0228	3-71
CRTC_PITCH	MMR:0x022C		IND:0x022C	3-74
OVR_CLR	MMR:0x0230		IND:0x0230	3-79
OVR_WID_LEFT_RIGHT	MMR:0x0234		IND:0x0234	3-79
OVR_WID_TOP_BOTTOM	MMR:0x0238		IND:0x0238	3-79
SNAPSHOT_VH_COUNTS	MMR:0x0240		IND:0x0240	3-215
SNAPSHOT_F_COUNT	MMR:0x0244		IND:0x0244	3-215
N_VIF_COUNT	MMR:0x0248		IND:0x0248	3-216
SNAPSHOT_VIF_COUNT	MMR:0x024C		IND:0x024C	3-216
CUR_OFFSET	MMR:0x0260		IND:0x0260	3-81
CUR_HORZ_VERT_POSN	MMR:0x0264		IND:0x0264	3-81
CUR_HORZ_VERT_OFF	MMR:0x0268		IND:0x0268	3-82
CUR_CLR0	MMR:0x026C		IND:0x026C	3-83
CUR_CLR1	MMR:0x0270		IND:0x0270	3-83
DAC_EXT_CNTL	MMR:0x0280		IND:0x0280	3-217
DAC_CRC_SIG	MMR:0x02CC		IND:0x02CC	3-126
DDA_CONFIG	MMR:0x02E0		IND:0x02E0	3-76
DDA_ON_OFF	MMR:0x02E4		IND:0x02E4	3-76
VGA_DDA_CONFIG	MMR:0x02E8		IND:0x02E8	3-77
VGA_DDA_ON_OFF	MMR:0x02EC		IND:0x02EC	3-77
OV0_COL_CONV	MMR:0x04FC		IND:0x04FC	3-217
PM4_BUFFER_DL_WPTR_DELAY	MMR:0x0718		IND:0x0718	3-218
CRC_CMDFIFO_ADDR	MMR:0x0740		IND:0x0740	3-218
CRC_CMDFIFO_DOUT	MMR:0x0744		IND:0x0744	3-219
PM4_VC_DEBUG_CONFIG	MMR:0x07A4		IND:0x07A4	3-219
PM4_VC_STAT	MMR:0x07A8		IND:0x07A8	3-221
PM4_VC_TIMESTAMP0	MMR:0x07B0		IND:0x07B0	3-221
PM4_VC_TIMESTAMP1	MMR:0x07B4		IND:0x07B4	3-221
VID_BUFFER_CONTROL	MMR:0x0900		IND:0x0900	3-222
BM_QUEUE_FREE_STATUS	MMR:0x0A14		IND:0x0A14	3-223
BM_ABORT	MMR:0x0A88		IND:0x0A88	3-224
SURFACE_DELAY	MMR:0x0B00		IND:0x0B00	3-224
SURFACE0_LOWER_BOUND	MMR:0x0B04		IND:0x0B04	3-225
SURFACE0_UPPER_BOUND	MMR:0x0B08		IND:0x0B08	3-226
SURFACE0_INFO	MMR:0x0B0C		IND:0x0B0C	3-228
SURFACE1_LOWER_BOUND	MMR:0x0B14		IND:0x0B14	3-225
SURFACE1_UPPER_BOUND	MMR:0x0B18		IND:0x0B18	3-226
SURFACE1_INFO	MMR:0x0B1C		IND:0x0B1C	3-229
SURFACE2_LOWER_BOUND	MMR:0x0B24		IND:0x0B24	3-225
SURFACE2_UPPER_BOUND	MMR:0x0B28		IND:0x0B28	3-226
SURFACE2_INFO	MMR:0x0B2C		IND:0x0B2C	3-230
SURFACE3_LOWER_BOUND	MMR:0x0B34		IND:0x0B34	3-225

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
SURFACE3_UPPER_BOUND	MMR:0x0B38		IND:0x0B38	3-226
SURFACE3_INFO	MMR:0x0B3C		IND:0x0B3C	3-231
AGP_CNTL_B	MMR:0x0B44		IND:0x0B44	3-231
VIPH_CH0_DATA	MMR:0x0C00		IND:0x0C00	3-213
VIPH_CH1_DATA	MMR:0x0C04		IND:0x0C04	3-214
VIPH_CH2_DATA	MMR:0x0C08		IND:0x0C08	3-214
VIPH_CH3_DATA	MMR:0x0C0C		IND:0x0C0C	3-214
DST_OFFSET	MMR:0x1404		IND:0x1404	3-137
DST_PITCH	MMR:0x1408		IND:0x1408	3-137
DST_WIDTH	MMR:0x140C		IND:0x140C	3-139
DST_HEIGHT	MMR:0x1410		IND:0x1410	3-139
SRC_X	MMR:0x1414		IND:0x1414	3-147
SRC_Y	MMR:0x1418		IND:0x1418	3-147
DST_X	MMR:0x141C		IND:0x141C	3-138
DST_Y	MMR:0x1420		IND:0x1420	3-138
SRC_PITCH_OFFSET	MMR:0x1428		IND:0x1428	3-146
DST_PITCH_OFFSET	MMR:0x142C		IND:0x142C	3-137
SRC_Y_X	MMR:0x1434		IND:0x1434	3-147
DST_Y_X	MMR:0x1438		IND:0x1438	3-139
DST_HEIGHT_WIDTH	MMR:0x143C		IND:0x143C	3-140
DP_GUI_MASTER_CNTL	MMR:0x146C		IND:0x146C	3-171
DP_BRUSH_BKGD_CLR	MMR:0x1478		IND:0x1478	3-164
DP_BRUSH_FRGD_CLR	MMR:0x147C		IND:0x147C	3-164
DST_WIDTH_X	MMR:0x1588		IND:0x1588	3-141
DST_HEIGHT_WIDTH_8	MMR:0x158C		IND:0x158C	3-140
SRC_X_Y	MMR:0x1590		IND:0x1590	3-147
DST_X_Y	MMR:0x1594		IND:0x1594	3-138
DST_WIDTH_HEIGHT	MMR:0x1598		IND:0x1598	3-140
DST_WIDTH_X_INCY	MMR:0x159C		IND:0x159C	3-141
DST_HEIGHT_Y	MMR:0x15A0		IND:0x15A0	3-141
DST_X_SUB	MMR:0x15A4		IND:0x15A4	3-144
DST_Y_SUB	MMR:0x15A8		IND:0x15A8	3-144
SRC_OFFSET	MMR:0x15AC		IND:0x15AC	3-146
SRC_PITCH	MMR:0x15B0		IND:0x15B0	3-146
DST_WIDTH_BW	MMR:0x15B4		IND:0x15B4	3-236
CLR_CMP_CNTL	MMR:0x15C0		IND:0x15C0	3-178
CLR_CMP_CLR_SRC	MMR:0x15C4		IND:0x15C4	3-178
CLR_CMP_CLR_DST	MMR:0x15C8		IND:0x15C8	3-178
CLR_CMP_MSK	MMR:0x15CC		IND:0x15CC	3-179
DP_SRC_FRGD_CLR	MMR:0x15D8		IND:0x15D8	3-164
DP_SRC_BKGD_CLR	MMR:0x15DC		IND:0x15DC	3-164

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
GUI_SCRATCH_REG0	MMR:0x15E0		IND:0x15E0	3-240
GUI_SCRATCH_REG1	MMR:0x15E4		IND:0x15E4	3-241
GUI_SCRATCH_REG2	MMR:0x15E8		IND:0x15E8	3-241
GUI_SCRATCH_REG3	MMR:0x15EC		IND:0x15EC	3-241
GUI_SCRATCH_REG4	MMR:0x15F0		IND:0x15F0	3-241
GUI_SCRATCH_REG5	MMR:0x15F4		IND:0x15F4	3-241
LEAD_BRES_ERR	MMR:0x1600		IND:0x1600	3-236
LEAD_BRES_INC	MMR:0x1604		IND:0x1604	3-236
LEAD_BRES_DEC	MMR:0x1608		IND:0x1608	3-237
TRAIL_BRES_ERR	MMR:0x160C		IND:0x160C	3-237
TRAIL_BRES_INC	MMR:0x1610		IND:0x1610	3-237
TRAIL_BRES_DEC	MMR:0x1614		IND:0x1614	3-237
TRAIL_X	MMR:0x1618		IND:0x1618	3-237
LEAD_BRES_LNTH	MMR:0x161C		IND:0x161C	3-236
TRAIL_X_SUB	MMR:0x1620		IND:0x1620	3-238
LEAD_BRES_LNTH_SUB	MMR:0x1624		IND:0x1624	3-238
DST_BRES_ERR	MMR:0x1628		IND:0x1628	3-142
DST_BRES_INC	MMR:0x162C		IND:0x162C	3-142
DST_BRES_DEC	MMR:0x1630		IND:0x1630	3-142
DST_BRES_LNTH	MMR:0x1634		IND:0x1634	3-142
DST_BRES_LNTH_SUB	MMR:0x1638		IND:0x1638	3-144
SC_LEFT	MMR:0x1640		IND:0x1640	3-155
SC_RIGHT	MMR:0x1644		IND:0x1644	3-155
SC_TOP	MMR:0x1648		IND:0x1648	3-155
SC_BOTTOM	MMR:0x164C		IND:0x164C	3-155
SRC_SC_RIGHT	MMR:0x1654		IND:0x1654	3-148
SRC_SC_BOTTOM	MMR:0x165C		IND:0x165C	3-148
AUX_SC_CNTL	MMR:0x1660		IND:0x1660	3-156
AUX1_SC_LEFT	MMR:0x1664		IND:0x1664	3-157
AUX1_SC_RIGHT	MMR:0x1668		IND:0x1668	3-157
AUX1_SC_TOP	MMR:0x166C		IND:0x166C	3-158
AUX1_SC_BOTTOM	MMR:0x1670		IND:0x1670	3-158
AUX2_SC_LEFT	MMR:0x1674		IND:0x1674	3-158
AUX2_SC_RIGHT	MMR:0x1678		IND:0x1678	3-158
AUX2_SC_TOP	MMR:0x167C		IND:0x167C	3-158
AUX2_SC_BOTTOM	MMR:0x1680		IND:0x1680	3-159
AUX3_SC_LEFT	MMR:0x1684		IND:0x1684	3-159
AUX3_SC_RIGHT	MMR:0x1688		IND:0x1688	3-159
AUX3_SC_TOP	MMR:0x168C		IND:0x168C	3-159
AUX3_SC_BOTTOM	MMR:0x1690		IND:0x1690	3-160
GUI_DEBUG0	MMR:0x16A0		IND:0x16A0	3-245
GUI_DEBUG1	MMR:0x16A4		IND:0x16A4	3-246

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
GUI_DEBUG2	MMR:0x16A8		IND:0x16A8	3-246
GUI_DEBUG3	MMR:0x16AC		IND:0x16AC	3-247
GUI_DEBUG4	MMR:0x16B0		IND:0x16B0	3-247
GUI_DEBUG5	MMR:0x16B4		IND:0x16B4	3-247
GUI_DEBUG6	MMR:0x16B8		IND:0x16B8	3-247
GUI_PROBE	MMR:0x16BC		IND:0x16BC	3-247
DP_CNTL	MMR:0x16C0		IND:0x16C0	3-165
DP_DATATYPE	MMR:0x16C4		IND:0x16C4	3-168
DP_MIX	MMR:0x16C8		IND:0x16C8	3-171
DP_WRITE_MSK	MMR:0x16CC		IND:0x16CC	3-171
DP_CNTL_XDIR_YDIR_YMAJOR	MMR:0x16D0		IND:0x16D0	3-170
DEFAULT_OFFSET	MMR:0x16E0		IND:0x16E0	3-243
DEFAULT_PITCH	MMR:0x16E4		IND:0x16E4	3-243
DEFAULT_SC_BOTTOM_RIGHT	MMR:0x16E8		IND:0x16E8	3-243
SC_TOP_LEFT	MMR:0x16EC		IND:0x16EC	3-161
SC_BOTTOM_RIGHT	MMR:0x16F0		IND:0x16F0	3-162
SRC_SC_BOTTOM_RIGHT	MMR:0x16F4		IND:0x16F4	3-148
FLUSH_1	MMR:0x1704		IND:0x1704	3-234
FLUSH_2	MMR:0x1708		IND:0x1708	3-234
FLUSH_3	MMR:0x170C		IND:0x170C	3-234
FLUSH_4	MMR:0x1710		IND:0x1710	3-234
FLUSH_5	MMR:0x1714		IND:0x1714	3-234
FLUSH_6	MMR:0x1718		IND:0x1718	3-235
FLUSH_7	MMR:0x171C		IND:0x171C	3-235
WAIT_UNTIL	MMR:0x1720		IND:0x1720	3-238
GUI_STAT	MMR:0x1740		IND:0x1740	3-244
PC_GUI_MODE	MMR:0x1744		IND:0x1744	3-251
PC_GUI_CTLSTAT	MMR:0x1748		IND:0x1748	3-253
PC_DEBUG_MODE	MMR:0x1760		IND:0x1760	3-256
HOST_DATA0	MMR:0x17C0		IND:0x17C0	3-151
HOST_DATA1	MMR:0x17C4		IND:0x17C4	3-151
HOST_DATA2	MMR:0x17C8		IND:0x17C8	3-151
HOST_DATA3	MMR:0x17CC		IND:0x17CC	3-152
HOST_DATA4	MMR:0x17D0		IND:0x17D0	3-152
HOST_DATA5	MMR:0x17D4		IND:0x17D4	3-153
HOST_DATA6	MMR:0x17D8		IND:0x17D8	3-153
HOST_DATA7	MMR:0x17DC		IND:0x17DC	3-153
HOST_DATA_LAST	MMR:0x17E0		IND:0x17E0	3-153
W_START	MMR:0x18CC		IND:0x18CC	3-250
SECONDARY_SCALE_PITCH	MMR:0x1980		IND:0x1980	3-160
SECONDARY_SCALE_X_INC	MMR:0x1984		IND:0x1984	3-160

Table A-3 MMR Registers Sorted by Address (Continued)

Register Name	Address	Secondary Adress	Indexed Address	Page
SECONDARY_SCALE_Y_INC	MMR:0x1988		IND:0x1988	3-161
SECONDARY_SCALE_HACC	MMR:0x198C		IND:0x198C	3-161
SECONDARY_SCALE_VACC	MMR:0x1990		IND:0x1990	3-161
MC_SRC2_CNTL	MMR:0x19D4		IND:0x19D4	3-261
MC_SRC1_CNTL	MMR:0x19D8		IND:0x19D8	3-261
MC_DST_CNTL	MMR:0x19DC		IND:0x19DC	3-261
MC_START_CNTL	MMR:0x19E0		IND:0x19E0	3-262
COMPOSITE_SHADOW_ID	MMR:0x1A0C		IND:0x1A0C	3-145
CLR_CMP_CLR_3D	MMR:0x1A24		IND:0x1A24	3-179
CLR_CMP_MSK_3D	MMR:0x1A28		IND:0x1A28	3-180
CONSTANT_COLOR	MMR:0x1A30		IND:0x1A30	3-251
Z_VIS	MMR:0x1AD4		IND:0x1AD4	3-251
WINDOW_XY_OFFSET	MMR:0x1BCC		IND:0x1BCC	3-249
DRAW_LINE_POINT	MMR:0x1BD0		IND:0x1BD0	3-249
DST_PITCH_OFFSET_C	MMR:0x1C80		IND:0x1C80	3-145
DP_GUI_MASTER_CNTL_C	MMR:0x1C84		IND:0x1C84	3-175
SC_TOP_LEFT_C	MMR:0x1C88		IND:0x1C88	3-162
SC_BOTTOM_RIGHT_C	MMR:0x1C8C		IND:0x1C8C	3-162
MISC_3D_STATE_CNTL_REG	MMR:0x1CA0		IND:0x1CA0	3-256
CONSTANT_COLOR_C	MMR:0x1D34		IND:0x1D34	3-260
PLANE_3D_MASK_C	MMR:0x1D44		IND:0x1D44	3-260

A.4 CGF Registers Sorted by Name

Table A-4 CGF Registers Sorted by Name

Register Name	Address	MMR Address	Indexed Address	Page
ADAPTER_ID	*CFG:0x002C*	*MMR:0x0F2C*	*IND:0x0F2C*	*3-7*
ADAPTER_ID_W	*CFG:0x004C*			*3-9*
AGP_COMMAND	*CFG:0x0058*	*MMR:0x0F58[R]*	*IND:0x0F58[R]*	*3-185*
AGP_STATUS	*CFG:0x0054*	*MMR:0x0F54*	*IND:0x0F54*	*3-185*
BASE_CODE	*CFG:0x000B*	*MMR:0x0F0B*	*IND:0x0F0B*	*3-5*
BIOS_ROM	*CFG:0x0030*	*MMR:0x0F30[R]*	*IND:0x0F30[R]*	*3-8*
BIST	*CFG:0x000F*	*MMR:0x0F0F*	*IND:0x0F0F*	*3-6*
CACHE_LINE	*CFG:0x000C*	*MMR:0x0F0C[R]*	*IND:0x0F0C[R]*	*3-5*
CAPABILITIES_ID	*CFG:0x0050*	*MMR:0x0F50*	*IND:0x0F50*	*3-9*
CAPABILITIES_PTR	*CFG:0x0034*	*MMR:0x0F34*	*IND:0x0F34*	*3-8*
COMMAND	*CFG:0x0004*	*MMR:0x0F04[R]*	*IND:0x0F04[R]*	*3-2*
DEVICE_ID	*CFG:0x0002*	*MMR:0x0F02*	*IND:0x0F02*	*3-2*
HEADER	*CFG:0x000E*	*MMR:0x0F0E*	*IND:0x0F0E*	*3-6*
INTERRUPT_LINE	*CFG:0x003C*	*MMR:0x0F3C[R]*	*IND:0x0F3C[R]*	*3-8*
INTERRUPT_PIN	*CFG:0x003D*	*MMR:0x0F3D*	*IND:0x0F3D*	*3-8*
IO_BASE	*CFG:0x0014*	*MMR:0x0F14[R]*	*IND:0x0F14[R]*	*3-7*
LATENCY	*CFG:0x000D*	*MMR:0x0F0D[R]*	*IND:0x0F0D[R]*	*3-5*
MAX_LATENCY	*CFG:0x003F*	*MMR:0x0F3F*	*IND:0x0F3F*	*3-9*
MEM_BASE	*CFG:0x0010*	*MMR:0x0F10[R]*	*IND:0x0F10[R]*	*3-6*
MIN_GRANT	*CFG:0x003E*	*MMR:0x0F3E*	*IND:0x0F3E*	*3-9*
PMI_CAP_ID	*CFG:0x005C*	*MMR:0x0F5C*	*IND:0x0F5C*	*3-189*
PMI_DATA	*CFG:0x0063*	*MMR:0x0F63*	*IND:0x0F63*	*3-190*
PMI_NXT_CAP_PTR	*CFG:0x005D*	*MMR:0x0F5D*	*IND:0x0F5D*	*3-189*
PMI_PMC_REG	*CFG:0x005E*	*MMR:0x0F5E*	*IND:0x0F5E*	*3-189*
PMI_PMCSR_REG	*CFG:0x0060*	*MMR:0x0F60[R]*	*IND:0x0F60[R]*	*3-190*
REG_BASE	*CFG:0x0018*	*MMR:0x0F18[R]*	*IND:0x0F18[R]*	*3-7*
REGPROG_INF	*CFG:0x0009*	*MMR:0x0F09*	*IND:0x0F09*	*3-5*
REVISION_ID	*CFG:0x0008*	*MMR:0x0F08*	*IND:0x0F08*	*3-4*
STATUS	*CFG:0x0006*	*MMR:0x0F06[R]*	*IND:0x0F06[R]*	*3-3*
SUB_CLASS	*CFG:0x000A*	*MMR:0x0F0A*	*IND:0x0F0A*	*3-5*
VENDOR_ID	*CFG:0x0000*	*MMR:0x0F00*	*IND:0x0F00*	*3-2*

A.5 CGF Registers Sorted by Address

Table A-5 CGF Registers Sorted by Address

Register Name	Address	MMR Address	Indexed Address	Page
VENDOR_ID	CFG:0x0000	MMR:0x0F00	IND:0x0F00	3-2
DEVICE_ID	CFG:0x0002	MMR:0x0F02	IND:0x0F02	3-2
COMMAND	CFG:0x0004	MMR:0x0F04[R]	IND:0x0F04[R]	3-2
STATUS	CFG:0x0006	MMR:0x0F06[R]	IND:0x0F06[R]	3-3
REVISION_ID	CFG:0x0008	MMR:0x0F08	IND:0x0F08	3-4
REGPROG_INF	CFG:0x0009	MMR:0x0F09	IND:0x0F09	3-5
SUB_CLASS	CFG:0x000A	MMR:0x0F0A	IND:0x0F0A	3-5
BASE_CODE	CFG:0x000B	MMR:0x0F0B	IND:0x0F0B	3-5
CACHE_LINE	CFG:0x000C	MMR:0x0F0C[R]	IND:0x0F0C[R]	3-5
LATENCY	CFG:0x000D	MMR:0x0F0D[R]	IND:0x0F0D[R]	3-5
HEADER	CFG:0x000E	MMR:0x0F0E	IND:0x0F0E	3-6
BIST	CFG:0x000F	MMR:0x0F0F	IND:0x0F0F	3-6
MEM_BASE	CFG:0x0010	MMR:0x0F10[R]	IND:0x0F10[R]	3-6
IO_BASE	CFG:0x0014	MMR:0x0F14[R]	IND:0x0F14[R]	3-7
REG_BASE	CFG:0x0018	MMR:0x0F18[R]	IND:0x0F18[R]	3-7
ADAPTER_ID	CFG:0x002C	MMR:0x0F2C	IND:0x0F2C	3-7
BIOS_ROM	CFG:0x0030	MMR:0x0F30[R]	IND:0x0F30[R]	3-8
CAPABILITIES_PTR	CFG:0x0034	MMR:0x0F34	IND:0x0F34	3-8
INTERRUPT_LINE	CFG:0x003C	MMR:0x0F3C[R]	IND:0x0F3C[R]	3-8
INTERRUPT_PIN	CFG:0x003D	MMR:0x0F3D	IND:0x0F3D	3-8
MIN_GRANT	CFG:0x003E	MMR:0x0F3E	IND:0x0F3E	3-9
MAX_LATENCY	CFG:0x003F	MMR:0x0F3F	IND:0x0F3F	3-9
ADAPTER_ID_W	CFG:0x004C			3-9
CAPABILITIES_ID	CFG:0x0050	MMR:0x0F50	IND:0x0F50	3-9
AGP_STATUS	CFG:0x0054	MMR:0x0F54	IND:0x0F54	3-185
AGP_COMMAND	CFG:0x0058	MMR:0x0F58[R]	IND:0x0F58[R]	3-185
PMI_CAP_ID	CFG:0x005C	MMR:0x0F5C	IND:0x0F5C	3-189
PMI_NXT_CAP_PTR	CFG:0x005D	MMR:0x0F5D	IND:0x0F5D	3-189
PMI_PMC_REG	CFG:0x005E	MMR:0x0F5E	IND:0x0F5E	3-189
PMI_PMCSR_REG	CFG:0x0060	MMR:0x0F60[R]	IND:0x0F60[R]	3-190
PMI_DATA	CFG:0x0063	MMR:0x0F63	IND:0x0F63	3-190

A.6 ATTR Registers Sorted by Name

Table A-6 ATTR Registers Sorted by Name

Register Name	Address	Page
ATTR00	*ATTR:0x0000*	*3-57*
ATTR10	*ATTR:0x0010*	*3-57*
ATTR11	*ATTR:0x0011*	*3-58*
ATTR12	*ATTR:0x0012*	*3-58*
ATTR13	*ATTR:0x0013*	*3-59*
ATTR14	*ATTR:0x0014*	*3-59*

A.7 CRT Regsiter Address Sorted by Name

Table A-7 CRT Regsiter Address Sorted by Name

Register Name	Address	Page
CRT00	CRT:0x0000	3-26
CRT00_S	CRT:0x0040	3-39
CRT01	CRT:0x0001	3-26
CRT01_S	CRT:0x0041	3-39
CRT02	CRT:0x0002	3-28
CRT02_S	CRT:0x0042	3-40
CRT03	CRT:0x0003	3-28
CRT03_S	CRT:0x0043	3-40
CRT04	CRT:0x0004	3-28
CRT04_S	CRT:0x0044	3-40
CRT05	CRT:0x0005	3-29
CRT05_S	CRT:0x0045	3-41
CRT06	CRT:0x0006	3-29
CRT06_S	CRT:0x0046	3-41
CRT07	CRT:0x0007	3-29
CRT07_S	CRT:0x0047	3-41
CRT08	CRT:0x0008	3-30
CRT08_S	CRT:0x0048	3-42
CRT09	CRT:0x0009	3-31
CRT09_S	CRT:0x0049	3-42
CRT0A	CRT:0x000A	3-31
CRT0A_S	CRT:0x004A	3-43
CRT0B	CRT:0x000B	3-32
CRT0B_S	CRT:0x004B	3-43
CRT0C	CRT:0x000C	3-32
CRT0C_S	CRT:0x004C	3-44
CRT0D	CRT:0x000D	3-32
CRT0D_S	CRT:0x004D	3-44
CRT0E	CRT:0x000E	3-33
CRT0E_S	CRT:0x004E	3-44
CRT0F	CRT:0x000F	3-33
CRT0F_S	CRT:0x004F	3-44
CRT10	CRT:0x0010	3-33
CRT10_S	CRT:0x0050	3-44
CRT11	CRT:0x0011	3-34
CRT11_S	CRT:0x0051	3-45
CRT12	CRT:0x0012	3-34
CRT12_S	CRT:0x0052	3-45
CRT13	CRT:0x0013	3-36

Table A-7 CRT Regsiter Address Sorted by Name (Continued)

Register Name	Address	Page
CRT13_S	CRT:0x0053	3-45
CRT14	CRT:0x0014	3-36
CRT14_S	CRT:0x0054	3-47
CRT15	CRT:0x0015	3-37
CRT15_S	CRT:0x0055	3-47
CRT16	CRT:0x0016	3-37
CRT16_S	CRT:0x0056	3-47
CRT17	CRT:0x0017	3-37
CRT17_S	CRT:0x0057	3-47
CRT18	CRT:0x0018	3-38
CRT18_S	CRT:0x0058	3-48
CRT1E	CRT:0x001E	3-38
CRT1E_S	CRT:0x005E	3-48
CRT1F	CRT:0x001F	3-39
CRT1F_S	CRT:0x005F	3-49
CRT22	CRT:0x0022	3-39
CRT22_S	CRT:0x0062	3-49

A.8 GRPH Registers Sorted by Name

Table A-8 GRPH Registers Sorted by Name

Register Name	Address	Page
GRA00	*GRPH:0x0000*	*3-50*
GRA01	*GRPH:0x0001*	*3-50*
GRA02	*GRPH:0x0002*	*3-51*
GRA03	*GRPH:0x0003*	*3-51*
GRA04	*GRPH:0x0004*	*3-52*
GRA05	*GRPH:0x0005*	*3-52*
GRA06	*GRPH:0x0006*	*3-53*
GRA07	*GRPH:0x0007*	*3-54*
GRA08	*GRPH:0x0008*	*3-54*

A.9 VGA Registers Sorted by Name

Table A-9 VGA Registers Sorted by Name

Register Name	Address	Secondary Address	Page
ATTRDR	*VGA_IO:0x03C1*		*3-56*
ATTRDW	*VGA_IO:0x03C0*		*3-56*
ATTRX	*VGA_IO:0x03C0*		*3-56*
CRTC8_DATA	*VGA_IO:0x03B5*	*VGA_IO:0x03D5*	*3-26*
CRTC8_IDX	*VGA_IO:0x03B4*	*VGA_IO:0x03D4*	*3-26*
DAC_DATA	*VGA_IO:0x03C9*		*3-19*
DAC_MASK	*VGA_IO:0x03C6*		*3-19*
DAC_R_INDEX	*VGA_IO:0x03C7*		*3-19*
DAC_W_INDEX	*VGA_IO:0x03C8*		*3-20*
GENENB	*VGA_IO:0x03C3*		*3-18*
GENFC_RD	*VGA_IO:0x03CA*		*3-16*
GENFC_WT	*VGA_IO:0x03BA*	*VGA_IO:0x03DA*	*3-16*
GENMO_RD	*VGA_IO:0x03CC*		*3-15*
GENMO_WT	*VGA_IO:0x03C2*		*3-14*
GENS0	*VGA_IO:0x03C2*		*3-17*
GENS1	*VGA_IO:0x03BA*	*VGA_IO:0x03DA*	*3-17*
GRPH8_DATA	*VGA_IO:0x03CF*		*3-50*
GRPH8_IDX	*VGA_IO:0x03CE*		*3-50*
SEQ8_DATA	*VGA_IO:0x03C5*		*3-21*
SEQ8_IDX	*VGA_IO:0x03C4*		*3-21*

A.10 SEQ Registers Sorted by Name

Table A-10 SEQ Registers Sorted by Name

Register Name	Address	Page
SEQ00	*SEQ:0x0000*	*3-21*
SEQ01	*SEQ:0x0001*	*3-23*
SEQ02	*SEQ:0x0002*	*3-24*
SEQ03	*SEQ:0x0003*	*3-24*
SEQ04	*SEQ:0x0004*	*3-25*

A.11 PLL Registers Sorted by Name

Table A-11 PLL Registers Sorted by Name

Register Name	Address	Page
AGP_PLL_CNTL	*PLL:0x0010*	*3-104*
CLK_PIN_CNTL	*PLL:0x0001*	*3-87*
FCP_CNTL	*PLL:0x0012*	*3-104*
HTOTAL_CNTL	*PLL:0x0009*	*3-97*
MCLK_CNTL	*PLL:0x000F*	*3-103*
MPLL_CNTL	*PLL:0x000E*	*3-102*
PLL_TEST_CNTL	*PLL:0x0013*	*3-105*
PPLL_CNTL	*PLL:0x0002*	*3-88*
PPLL_DIV_0	*PLL:0x0004*	*3-90*
PPLL_DIV_1	*PLL:0x0005*	*3-91*
PPLL_DIV_2	*PLL:0x0006*	*3-92*
PPLL_DIV_3	*PLL:0x0007*	*3-93*
PPLL_REF_DIV	*PLL:0x0003*	*3-89*
VCLK_ECP_CNTL	*PLL:0x0008*	*3-94*
X_MPLL_REF_FB_DIV	*PLL:0x000A*	*3-98*
XCLK_CNTL	*PLL:0x000D*	*3-101*
XDLL_CNTL	*PLL:0x000C*	*3-99*
XPLL_CNTL	*PLL:0x000B*	*3-98*

This page intentionally left blank.

Appendix B
Revision History

B.1 P/N RRG-G04500-C, Rev 0.01 (RR45001C.pdf)

This document complies with Eng. RAGE 128 PRO Register Document REV 1.1 (Dec. 21, 1998). It is based on the RAGE 128 VR/GL Registers specification, and it includes the following changes.

The following registers were created:

- AGP_CNTL_B
- AMCGPIO_A_MIR
- AMCGPIO_EN_MIR
- AMCGPIO_MASK_MIR
- AMCGPIO_Y_MIR
- AUX_WINDOW_HORZ_CNTL
- AUX_WINDOW_VERT_CNTL
- DAC_EXT_CNTL
- FP_CRTC_H_TOTAL_DISP
- FP_CRTC_V_TOTAL_DISP
- FP_GEN_CNTL
- FP_HORZ_STRETCH
- FP_H_SYNC_STRT_WID
- FP_PANEL_CNTL
- FP_VERT_STRETCH
- FP_V_SYNC_STRT_WID
- HW_DEBUG2
- OV0_COL_CONV
- PAD_AGPINPUT_DELAY
- PAD_CTLR_STRENGTH
- PAD_CTLR_UPDATE
- PC_MISC_CTL
- PM4_VC_DEBUG_CONFIG
- PM4_VC_STAT
- PM4_VC_TIMESTAMP0
- PM4_VC_TIMESTAMP1

- SEPROM_CNTL
- TMDS_CNTL
- TMDS_CRC
- TMDS_PLL_CNTL
- TMDS_SYNC_CHAR_SETA
- TMDS_SYNC_CHAR_SETB
- TMDS_TRANSMITTER_CNTL
- Z_VIS

The following registers were modified:

- AGP_COMMAND
- AGP_PLL_CNTL
- AGP_STATUS
- AMCGPIO_A_REG
- AMCGPIO_EN_REG
- AMCGPIO_MASK
- AMCGPIO_Y_REG
- CAP0_CONFIG
- CAP0_DWNSC_XRATIO
- CAP1_CONFIG
- CAP1_DWNSC_XRATIO
- CAP_INT_STATUS
- CRT00_S
- CRT04_S
- CRT05_S
- CRT06_S
- CRT07_S
- CRT09
- CRT0A
- CRT0B
- CRT10_S
- CRT11_S
- CRT14
- CRTC_EXT_CNTL
- CRTC_STATUS
- DAC_CNTL

- GUI_DEBUG0
- MEM_CNTL
- MPLL_CNTL
- MPP_GP_STROBE_SEQ
- OV0_SCALE_CNTL
- PLL_TEST_CNTL
- PM4_CMDFIFO_DATAH
- PM4_VC_CNTL
- PPLL_CNTL
- TEST_DEBUG_MUX
- TEX_PALETTE_WR_INDEX
- TRI_LIN_CNTL
- VIDEOMUX_CNTL
- XCLK_CNTL
- XPLL_CNTL
- Z_OFFSET
- Z_PITCH

The following registers were deleted:

- CAP0_XSHARPNESS
- CAP1_XSHARPNESS
- GPIO_MONIDB
- MPP_TB_ADDR
- MPP_TB_CONFIG
- MPP_TB_DATA
- MPP_TB_STROBE_SEQ

B.2 P/N RRG-G04500-C, Rev 0.02 (RR45002C.pdf)

This document complies with Eng. RAGE 128 PRO Register Document REV 1.95 (Feb. 12, 1999).The changes are as follows.

The following register was created:

- PM4_BUFFER_DL_WPTR_DELAY

The following registers were modified:

- CAP0_CONFIG
- CAP1_CONFIG

B.3 Mar. 1999: P/N RRG-G04500-C, Rev 0.03 (RR45003C.pdf)

This document now complies with the current engineering specifications. The register address is shown in brackets.

The following registers were added:

- DP_GUI_MASTER_CNTL (MMR-1C84)
- FOG_3D_TABLE_DENSITY (MMR-1818)
- FOG_3D_TABLE_END (MMR-1814)
- FOG_3D_TABLE_START (MMR-1810)
- FOG_COLOR (MMR-1A10)
- FOG_COLOR_C (MMR-1CAC)
- FOG_TABLE_DATA (MMR-1A18)
- FOG_TABLE_INDEX (MMR-1A14)
- HOST_DATA0 (MMR-17C0)
- HOST_DATA1 (MMR-17C4)
- HOST_DATA2 (MMR-17C8)
- HOST_DATA3 (MMR-17CC)
- HOST_DATA4 (MMR-17D0)
- HOST_DATA5 (MMR-17D4)
- HOST_DATA6 (MMR-17D8)
- HOST_DATA7 (MMR-17DC)
- HOST_DATA_LAST (MMR-17E0)
- OV0_P1_V_ACCM_INIT (MMR-0428)
- OV0_P23_BLANK_LINES_AT_TOP (MMR-0434)
- OV0_SCALE_CNTL (MMR-0420)
- OV0_V_INC (MMR-0424)
- PPLL_DIV_0 (PLL-04)
- PPLL_DIV_1 (PLL-05)
- PPLL_DIV_2 (PLL-06)
- PPLL_DIV_3 (PLL-07)
- TEX_CACHE_STAT_COUNT (MMR-1974)
- SC_BOTTOM_RIGHT_C (MMR-1C8C)
- SC_TOP_LEFT_C (MMR-1C88)

- SCALE_SCR_HEIGHT_WIDTH (MMR-1994)
- SCALE_OFFSET_0 (MMR-1998)
- SCALE_PITCH (MMR-1998)
- SCALE_X_INC (MMR-19A0)
- SCALE_Y_INC (MMR-19A4)
- SCALE_HACC (MMR-19A8)
- SCALE_VACC (MMR-19AC)
- SCALE_DST_X_Y (MMR-19B0)
- SCALE_DST_HEIGHT_WIDTH (MMR-19B4)
- SCALE_3D_CNTL (MMR-1A00)
- SCALE_3D_DATATYPE (MMR-1A20)
- SC_TOP_LEFT_C (MMR-1C88)
- VERTEX_1_SPEC_ARGB (MMR-1B90)

The following registers had their titles corrected:

- PRIM_7_OFFSET_C is now referred to as PRIM_TEX_7_OFFSET_C
- CAP0_BUF0_OFFSETC is now referred to as CAP0_BUF0_OFFSET
- GIU_SCRATCH_REG0 is now referred to as GUI_SCRATCH_REG0
- GIU_SCRATCH_REG1 is now referred to as GUI_SCRATCH_REG1
- GIU_SCRATCH_REG2 is now referred to as GUI_SCRATCH_REG2
- GIU_SCRATCH_REG3 is now referred to as GUI_SCRATCH_REG3
- GIU_SCRATCH_REG4 is now referred to as GUI_SCRATCH_REG4
- GIU_SCRATCH_REG5 is now referred to as GUI_SCRATCH_REG5
- REGPROG_ID is now referred to as REGPROG_INF
- VGA_DDA_0N_OFF is now referred to as VGA_DDA_ON_OFF

The following registers had their addresses corrected:

- SNAPSHOT_VIF_COUNT was MMR-02C4 and is now MMR-024C
- CAP1_VBI_EVEN_OFFSET was MMR-093C and is now MMR-09B0
- I2C_DATA was MMR-0090 and is now MMR-0098
- INTERRUPT_LINE was MMR-003C and is now MMR-0F3C
- PMI_CAP_ID was MMR-0000 and is now MMR-0F5C
- SNAPSHOT_VIF_COUNT was MMR-00C4 and is now MMR-024C

- VERTEX_2_Z was MMR-1B6C and is now MMR-1BA0
- VIPH_CH0_DATA was MMR-01E4 and is now MMR-0C00
- VIPH_CH1_DATA was MMR-01E8 and is now MMR-0C04
- VIPH_CH2_DATA was MMR-01EC and is now MMR-0C08
- VIPH_CH3_DATA was MMR-01F0 and is now MMR-0C0C
- Z_STEN_CNTL_C was MMR-1AC4 and is now MMR-1C98

The following register was removed:

- PRIM_6_OFFSET_C (MMR-1928)

B.4 May 1999: P/N RRG-G04500-C, Rev 0.04 (RR45004C.pdf)

This document complies with Eng. RAGE 128 PRO Register Document REV 1.98 (Apr. 10, 1999).The changes are as follows.

New registers:

- PM4_BUFFER_DL_WPTR_DELAY (MMR-0718)

Modified registers:

- AGP_CNTL_B (MMR-0B444)
- AMCGPIO_A_MIR (MMR-00A0)
- AMCGPIO_EN_MIR (MMR-00A8)
- AMCGPIO_MASK_MIR (MMR-009C)
- AMCGPIO_Y_MIR (MMR-00A4)
- AUX_WINDOW_HORZ_CNTL (MMR-02D8)
- AUX_WINDOW_VERT_CNTL (MMR-02DC)
- FP_GEN_CNTL (MMR-0284)
- SEPROM_CNTL (MMR-006C)
- AMCGPIO_A_REG (MMR-01A0)
- AMCGPIO_EN_REG (MMR-01A8)
- AMCGPIO_MASK (MMR-0194)
- AMCGPIO_Y_REG (MMR-01A4)
- BM_CHUNK_0_VAL (MMR-0A18)
- BUS_CNTL1 (MMR-0034)
- COMPOSITE_SHADOW_ID (MMR-1A0C)

- CONFIG_BONDS (MMR-00E8)
- CONFIG_XSTRAP (MMR-00E4)
- CRT09_S (CRT-49)
- CRT14 (CRT-14)
- GEN_INT_CNTL (MMR-0040)
- GEN_INT_STATUS (MMR-0044)
- HW_DEBUG (MMR-0128)
- MEM_ADDR_CONFIG (MMR-0148)
- MEM_INTF_CNTL (MMR-014C)
- MEM_STR_CNTL (MMR-0150)
- MPP_GP_STROBE_SEQ (MMR-01CC)
- OV0_SCALE_CNTL (MMR-0420)
- OV0_TEST (MMR-04F8)
- PC_NGUI_MODE (MMR-0180)
- PM4_CMDFIFO_ADDR (MMR-07E4)
- PM4_FPU_STAT (MMR-07A0)
- PM4_VC_FPU_SETUP(MMR-071C)
- PRIMARY_TEXTURE_COMBINE_CNTL (MMR-1A08)
- PRIM_TEXTURE_COMBINE_CNTL_C (MMR-1CB4)
- SCALE_3D_CNTL (MMR-1A00)
- SCALE_3D_DATATYPE (MMR-1A20)
- SECONDARY_TEXTURE_COMBINE_CNTL (MMR-1A34)
- SEC_TEX_COMBINE_CNTL_C (MMR-1D04)
- SETUP_CNTL (MMR-1BC4)
- SETUP_CNTL_PM4 (MMR-1BD4)
- TEST_DEBUG_CNTL (MMR-0120)
- TEX_CNTL (MMR-1800)
- TEX_CNTL_C (MMR-1C9C)
- Z_OFFSET (MMR-1AC4)
- GENMO_WT (VGA_IO-3C2): fixed bits 6 and 7.
- GENMO_RD (VGA_IO-3CC): fixed bits 6 and 7.

The following registers were deleted:

- The following registers were deleted:
- CAP0_XSHARPNESS (MMR-097C)
- CAP1_XSHARPNESS (MMR-09EC)
- GPIO_MONIDB (MMR-006C)
- MPP_TB_ADDR (MMR-0070)
- MPP_TB_CONFIG (MMR-01C0)
- MPP_TB_DATA (MMR-0074)
- MPP_TB_STROBE_SEQ (MMR-01C4)

B.5 June 1999: P/N RRG-G04500-C, Rev 0.05 (RR45005C.pdf)

- This document complies with Eng Rev. Level 1.103 as of June 22, 1999.

B.6 Jan 2000: P/N RRG-G04500-C, Rev 1.01 (RR45101C.pdf)

The following registers have been modified:

- HW_DEBUG (HW_7_DEBUG and HW_9_DEBUG.
- PMI_PMC_REG (PMI_VERSION).
- HW_DEBUG (HW_4_DEBUG and HW_5_DEBUG).
- AGP_CNTL_B (AGP_E_MISC and AGP_F_MISC).
- DEVICE_ID

The following registers have been removed:

- FOG_3D_TABLE_START
- FOG_3D_TABLE_END
- FOG_3D_TABLE_DENSITY
- DESTINATION_3D_CLR_CMP_VAL
- DESTINATION_3D_CLR_CMP_MSK