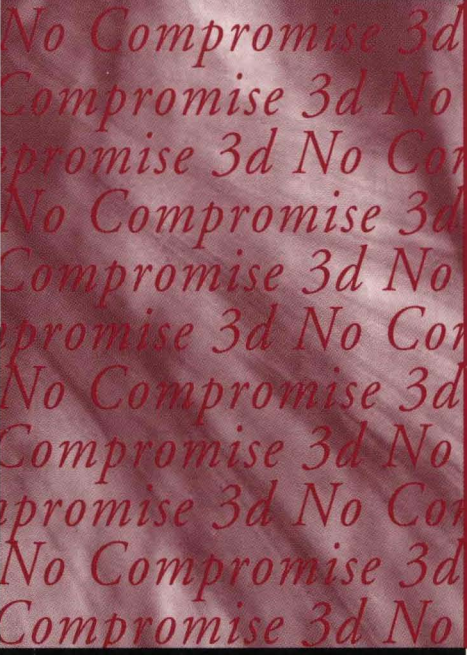


VIRGE/VX

ViRGE

s3d

ViRGE/VX Integrated 3D Accelerator

June 1996

S3 Incorporated
2770 San Tomas Expressway
P.O. Box 58058
Santa Clara, CA 95052-8058

NOTATIONAL CONVENTIONS

The following notational conventions are used in this data book:

Signal names are shown in all uppercase letters. For example, XD.

A bar over a signal name indicates an active low signal. For example, $\overline{OE}$.

n-m indicates a bit field from bit n to bit m. For example, 7-0 specifies bits 7 through 0, inclusive.

n:m indicates a signal (pin) range from n to m. For example D[7:0] specifies data lines 7 through 0, inclusive

Use of a trailing letter H indicates a hexadecimal number. For example, 7AH is a hexadecimal number.

Use of a trailing letter b indicates a binary number. For example, 010b is a binary number.

When K or M are used, they refer to binary rather than decimal form. Thus, for example, 1 KByte would be equivalent to 1024, not 1,000 bytes.

When k is used, it refers to decimal 1000.

NOTICES

Additional information may be obtained from:

S3 Incorporated, Literature Department, P.O. Box 58058, Santa Clara, CA 95052-8058.

Telephone: 408-980-5400, Fax: 408-980-5444

Table of Contents

List of Figures

List of Tables

Section 1: Introduction

High-Performance VRAM-based 2D/3D Graphics and Video Accelerator

- High-performance 64-bit 2D/3D graphics engine
- Integrated 220 MHz RAMDAC and clock synthesizer
- S3 Streams Processor for accelerated video
- S3 Scenic Highway for direct interface to live video and MPEG-1 peripherals

S3d Graphics Engine Features

- High performance 2D Windows acceleration
- Flat and Gouraud shading for 3D
- High quality/performance 3D texture mapping
- Perspective correction
- Bi-linear and tri-linear texture filtering
- MIP-Mapping
- Depth cueing and fogging
- Alpha blending
- Video texture mapping
- Z-buffering

S3 Streams Processor Features

- Supports on-the-fly stretching and blending of primary RGB stream and RGB or YUV (video) secondary stream
- Each stream can have a different color depth
- High-quality hardware-assisted video playback
- Horizontal and vertical interpolation
- Support for Indeo, Cinepak, and software and hardware-accelerated MPEG-1 video

S3 Scenic Highway Interface

- Philips SAA7110/SAA7111 video digitizers
- S3 Scenic/MX2 MPEG-1 audio/video decoder

High Resolution Support Up to 1600x1200x24

High-Performance Memory Support

- 64-bit VRAM memory interface with Block write support
- Two independent 64-bit pixel data busses
- 2-, 4- or 8-MBytes of video memory
- Single-cycle EDO operation

Non-x86 CPU Support

- Big endian/little endian byte ordering
- Relocatable addressing
- Packed 24 bits/pixel memory addressing with alpha pitching

Industry-Standard Local Bus Support

- Glueless PCI 2.1 bus interface

PCI Bus Mastering for Display List Processing and Video Capture Support

Multimedia Support Hooks

- S3 Scenic Highway
- VESA® advanced feature connector
- 8- and 16-bit bi-directional feature connector

Full Software Support

- Drivers for major operating systems: [Windows® 95, Windows™ 3.11, Windows NT™, OS/2® 2.1 and 3.0 (Warp™), ADI 4.2]

Green PC/Monitor Plug and Play Support

- Full hardware and BIOS support for VESA Display Power Management Signaling (DPMS) monitor power savings modes
- DDC monitor communications

Extensive Static/Dynamic Power Management

388-pin BGA package

1.1 OVERVIEW

The S3® ViRGE/VX™ integrated 3D video/graphics accelerator (hereinafter referred to as ViRGE/VX) enables development of compelling interactive entertainment, education, and presen-tation applications for the mainstream personal computing world. It also provides the highest performance for high-end desktop applications.

1.2 64-bit S3d ENGINE

The ViRGE/VX S3d™ Engine provides 2D acceleration for excellent Windows applications performance and a full-featured high-performance 3D rendering engine for games and other 3D applications.

The S3d Engine incorporates the key Windows accelerator functions of BitBLT, line draw and polygon fill. 3D features include flat shading, Gouraud shading and texture mapping support. Advanced texture mapping features include perspective correction, bi-linear and tri-linear filtering, MIP-Mapping, and Z-buffering. The S3d Engine also includes direct support for utilizing a video as a texture map. These features provide the most realistic user experience for interactive 3D applications.

Other advanced features of the S3d Engine include S3 proprietary compressed texture formats

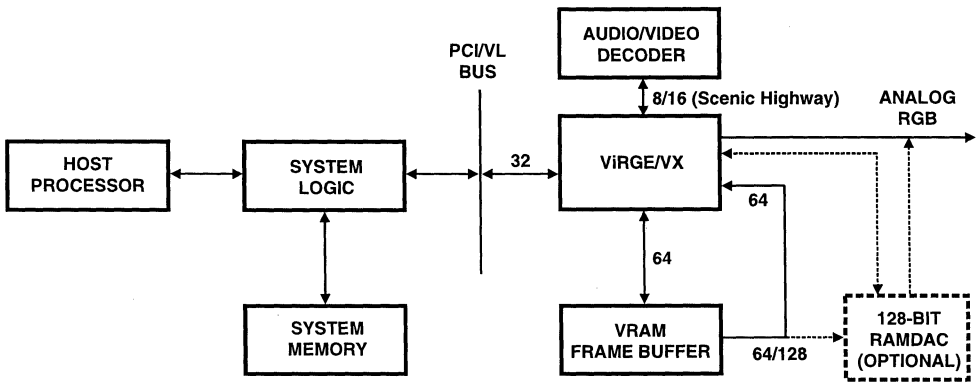


Figure 1-1. System Block Diagram

for improved performance and reduced memory requirements, as well as support for S3's MUX buffering (pat. pend.) feature, which allows for Z-buffering support with no additional memory requirement.

1.3 Streams Processor

The S3 Streams Processor™ provides the stretching and YUV color space conversion features required for full screen video playback with both software CODECs and hardware MPEG-1 sources. The Streams Processor allows simultaneous display of graphics and video of different color depths.

1.4 S3 Scenic Highway

The S3 Scenic Highway™ allows lowest cost direct connection to S3's Scenic/MX2™ MPEG-1 audio and video decoder as well as video digitizers such as Philips®, SAA7110/SAA7111.

1.5 MORE INFORMATION

For more detailed information about programming for the ViRGE/VX product, contact your local S3 representative or S3 directly for a copy of the *S3 ViRGE 3D Accelerator Software User's Guide*.

1.6 VIDEO RESOLUTIONS SUPPORTED

Table 1-1. Video Resolutions Supported

				Maximum Vertical Refresh Rate (Hz)		
Resolution	**2 MB**	**4 MB**	**6MB**	**175 MHz Part (Internal DAC)**	**220 MHz Part (Int/Ext DAC)**	**Streams Processor Active (135 MHz Max)**
640x480x4	✔	✔	✔	404	508	312
640x480x8	✔	✔	✔	404	508	312
640x480x16	✔	✔	✔	404	508	312
640x480x24	✔	✔	✔	312	508	312
800x600x4	✔	✔	✔	259	325	200
800x600x8	✔	✔	✔	259	325	200
800x600x16	✔	✔	✔	259	325	200
800x600x24	✔	✔	✔	200	325	200
1024x768x4	✔	✔	✔	158	199	122
1024x768x8	✔	✔	✔	158	199	122
1024x768x16	✔	✔	✔	158	199	122
1024x768x24		✔	✔	122	199	122
1280x1024x4	✔	✔	✔	95	119	73
1280x1024x8	✔	✔	✔	95	119	73
1280x1024x16		✔	✔	95	119	73
1280x1024x24		✔	✔	73	119	73
1600x1200x4	✔	✔	✔	65	81	50
1600x1200x8	✔	✔	✔	65	81	50
1600x1200x16		✔	✔	65	81	50
1600x1200x24*			✔	-	81	-

Section 2: Mechanical Data

2.1 THERMAL SPECIFICATIONS

Parameter	Min	Typ	Max	Unit
Thermal Corellation Coefficient Ψ_{JT}		0.1		°C/W
Thermal Resistance Θ_{JA} (Still Air)		11.3		°C/W
Junction Temperature			125	°C

2.2 MECHANICAL DIMENSIONS

ViRGE/VX comes in a 388-pin BGA package. The mechanical dimensions are given in Figure 2-1.

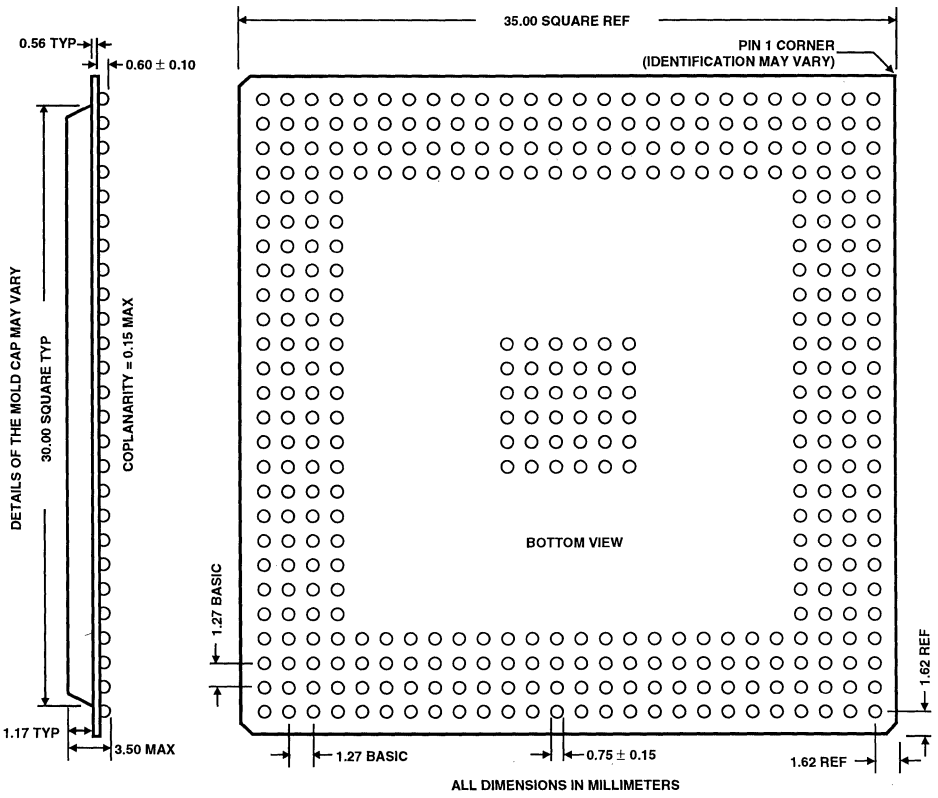


Figure 2-1. 388-pin BGA Mechanical Dimensions

Section 3: Pins

3.1 PINOUT DIAGRAMS

ViRGE/VX comes in a 388-pin BGA package. The pin locations are shown in Figure 3-1.

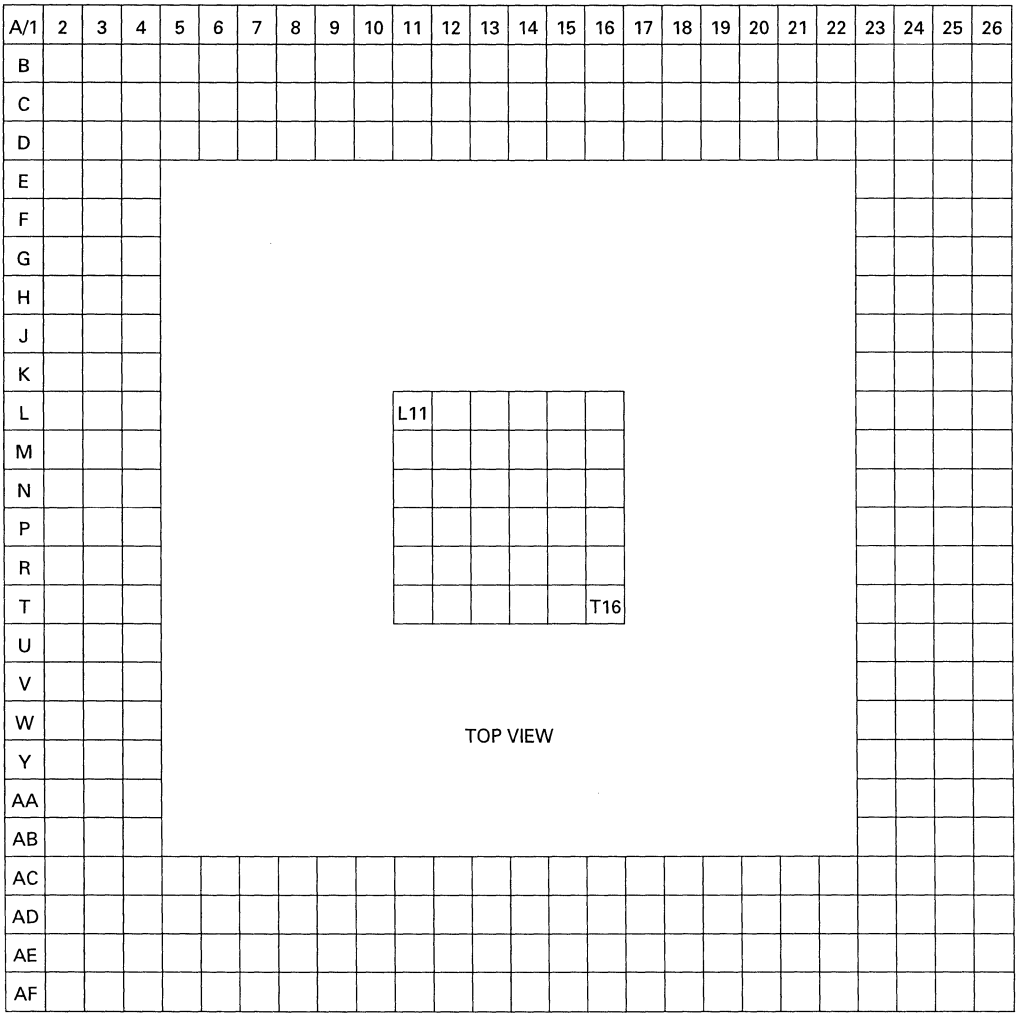


Figure 3-1. Pin Locations

3.2 PIN DESCRIPTIONS

The following table provides a brief description of each pin. The following abbreviations are used for pin types.

I - Input signal
O - Output signal
B - Bidirectional signal

Table 3-1. Pin Descriptions

Symbol	Type	Description
PCI BUS INTERFACE		+
Address and Data		
AD[31:0]	B	Multiplexed Address/Data Bus. A bus transaction (cycle) consists of an address phase followed by one or more data phases.
$\overline{C/BE[3:0]}$	B	Bus Command/Data Byte Enables. These signals carry the bus command during the address phase and the byte enables during the data phase. These signals are outputs during bus master DMA operation.
Bus Control		
SCLK	I	PCI System Clock.
$\overline{INTA}$	O	Interrupt Request.
$\overline{IRDY}$	B	Initiator Ready. A bus data phase is completed when both $\overline{IRDY}$ and $\overline{TRDY}$ are asserted on the same cycle. This signal is an output during bus master DMA operation.
$\overline{TRDY}$	B	Target Ready. A bus data phase is completed when both $\overline{IRDY}$ and $\overline{TRDY}$ are asserted on the same cycle. The signal is an input for bus master DMA operation.
$\overline{DEVSEL}$	B	Device Select. ViRGE/VX drives this signal active when it decodes its address as the target of the current access. This signal is an input for bus master DMA operation.
IDSEL	I	Initialization Device Select. This input is the chip select for PCI configuration register reads/writes.
$\overline{RESET}$	I	System Reset. Asserting this signal forces the registers and state machines to a known state.
$\overline{FRAME}$	B	Cycle Frame. This signal is asserted by the bus master to indicate the beginning of a bus transaction. It is deasserted during the final data phase of a bus transaction. This signal is an output for bus master DMA operation.
PAR	B	Parity. ViRGE/VX asserts this signal to verify even parity during reads. When scan testing is enabled, data is output on this pin. This signal is an input for bus master DMA operation.
$\overline{STOP}$	B	Stop. ViRGE/VX asserts this signal to indicate a target disconnect. This signal is an input for bus master DMA operation.
$\overline{REQ}$	O	Request. ViRGE/VX asserts this signal to request control of the PCI bus for bus master DMA operation.
$\overline{GNT}$	I	Grant. When asserted, this signal indicates that ViRGE/VX has been granted control of the PCI bus for bus master DMA operation.

Table 3-1. Pin Descriptions (Continued)

Symbol	Type	Description
CLOCK CONTROL		
XIN	I	Reference Frequency Input. If an external crystal is used, it is connected between XOUT and this pin. A stable external frequency source can also be input via this pin.
XOUT	O	Crystal Output. If an external 14.318 MHz crystal is used, it is connected between XIN and this pin. This pin drives the crystal via an internal oscillator.
DISPLAY MEMORY INTERFACE		
Address and Data		
MA[8:0]	O	Memory Address Bus. The video memory row and column addresses are multiplexed on these lines.
PD[63:0]	B	Display Memory Pixel Data Bus Lines. PD[28:0] are also used as the system configuration strapping bits, providing system configuration and setup information upon power-on or reset. After reset, the ROM address and data signals are multiplexed on 24 of these pins.
SID[63:0]	I	Serial Interface Data. Serial memory data from the VRAM when the internal RAMDAC is used.
Memory Control		
$\overline{\text{RAS}}$[1:0]	O	Row Address Strobes.
$\overline{\text{CAS}}$[7:0]	O	Column Address Strobes.
$\overline{\text{WE}}$[1:0]	O	Write Enables.
$\overline{\text{OE}}$[1:0]	O	Output Enables.
SC[3:0]	O	Serial (shift) Clock. These outputs clock the serial out data from the VRAM.
$\overline{\text{SE}}$[3:0]	O	Serial Enable. These signals enable serial output from the VRAM.
DSF	O	Special VRAM Function Control.

Table 3-1. Pin Descriptions (Continued)

Symbol	Type	Description
VIDEO INTERFACE		
COMP	I	Compensation Pin. This pin is tied to VDDA through a 0.1 µF capacitor.
PDOWN	I	Power Down. Asserting this signal turns off the RGB analog output from the DACs. If PD11 is strapped low at power-on, this becomes the DCLK (dot clock) input, bypassing the internal oscillator. This is normally only used for test purposes.
VREF		Voltage Reference. This pin is tied to V_{SS} through a 0.1 µF capacitor.
RSET		Reference Resistor. This pin is tied to V_{SS} through an external resistor to control the full-scale current value.
AR	O	Analog Red. Analog red output to the monitor.
AG	O	Analog Green. Analog green output to the monitor.
AB	O	Analog Blue. Analog blue signal to the monitor.
ENFEAT	O	Enable Feature Connector. Setting SRD_0 to 1 drives this signal low when SR1C_1-0 are 00b. This also enables all feature connector operations. If PD11 is strapped low at power-on, this becomes the MCLK (dot clock) input, bypassing the internal oscillator. This is normally only used for test purposes. This external MCLK may also be used for clocking during scan testing.
BLANK	B	Video Blank. When ESYNC is high, BLANK is a feature connector output. When ESYNC is low, BLANK is a feature connector input that, when driven low, turns off the video output.
ESYNC	I	External SYNC. When ESYNC is driven low, HSYNC, VSYNC and BLANK become inputs. When ESYNC is high, HSYNC, VSYNC and BLANK become outputs.
EVIDEO	I	External Video. When this input is driven low, PA[15:0] are inputs and are sampled by VCLKI. When this input is high, the PA signals are outputs to the feature connector.
EVCLK	I	External VCLK. When this input is asserted low, VCLK is an input to the internal RAMDAC. When this input is high, VCLK is output to the feature connector.
VCLK	B	Video/Pixel Clock. When EVCLK is high, this signal is an output to the feature connector. When EVCLK is low, this becomes an input used only for test purposes.
VCLKI	I	VCLK Input. The VCLKI function is enabled when LPB VAFC (16-bit) feature connector operation is enabled. Setting bit 1 of SRB to 1 causes VCLKI to be used to clock in feature connector pixel data to the internal RAMDAC.
HSYNC	B	Horizontal Sync. When ESYNC is high, this is the horizontal sync output. When ESYNC is low, this is an input from the feature connector.
VSYNC	B	Vertical Sync. When ESYNC is high, this is the vertical sync output. When ESYNC is low, this is an input from the feature connector.
PA[15:0]	B	Pixel Address Lines [15:0]. When EVIDEO is high, PA signals are outputs to the feature connector. When EVIDEO is low, PA signals are inputs and are sampled by VCLKI if bit 1 of SRB is set to 1.

Table 3-1. Pin Descriptions (Continued)

Symbol	Type	Description
EXTERNAL RAMDAC INTERFACE		
$\overline{\text{DACRD}}$	O	DAC Read.
$\overline{\text{DACWR}}$	O	DAC Write
DACSCLK	I	DAC SCLK. This input is returned from the RAMDAC and is used as the internal ICLK.
DACLCLK	O	DAC LCLK. This clocks pixel data to the RAMDAC and is the same as the SC[3:0] frequency.
DACD[7:0]	B	DAC Data. RAMDAC register read/write data.
RS[3:0]	O	Register Select.
MISCELLANEOUS FUNCTIONS		
General Data, I/O and Serial Ports		
RA[15:0]	O	ROM Address Bus. These signals provide the address for BIOS ROM reads. They are multiplexed with PD signals. Programmers must ensure that the memory bus is inactive when reading the ROM.
RD[7:0]	I	ROM Data Bus. These signals carry data for BIOS ROM reads. They are multiplexed with PD signals. Programmers must ensure that the memory bus is inactive when reading the ROM.
$\overline{\text{ROMEN}}$	O	ROM Enable. This signal provides the chip output enable input for BIOS ROM reads.
GOP[1:0]	O	General Output Port Bits 1-0. If SR1C_1 is set to 1, the value of CR5C_0 is output on pin A16 (GOP0) and the value of CR5C_1 is output on pin B8 (GOP1).
$\overline{\text{STWR}}$	O	Strobe Write. If SR1C_1 is cleared to 0, this signal is asserted whenever a write is made to CR5C. It is used to enable a General Output Port latch.
SPCLK	B	Serial Port Clock. This is the clock for serial data transfer, either for I^2C or DDC2 monitor data communications. As an output, it is programmed via MMFF20_0. As an input, its status is read via MMFF20_2. In either case the serial port must be enabled by setting MMFF20_4 to 1. PD[26:25] can be strapped to allow I/O (E2H or E8H) access to the Serial Port register while ViRGE/VX is disabled.
SPD	B	Serial Port Data. This is the data signal for serial data transfer, either for I^2C or DDC2 monitor data communications. As an output, it is programmed via MMFF20_1. As an input, its status is read via MMFF20_3. In either case the serial port must be enabled by setting MMFF20_4 to 1. PD[26:25] can be strapped to allow I/O (E2H or E8H) access to MMFF20 while ViRGE/VX is disabled.

Table 3-1. Pin Descriptions (Continued)

Symbol	Type	Description
LOCAL PERIPHERAL BUS		
Scenic/MX2 Mode		
LD[7:0]	B	LPB Data. This is the Scenic Highway data bus and carries compressed data to the Scenic/MX2 and video data from the Scenic/MX2.
LCLK	I	LPB Clock. This clock controls transactions between ViRGE/VX and Scenic Highway peripherals
$\overline{\text{VREQ}}$/VRDY	O	Video Request/Ready. This signal is part of the the Scenic Highway data transfer protocol between ViRGE/VX and the Scenic/MX2.
$\overline{\text{CREQ}}$/CRDY	I	Scenic/MX2 Request/Ready. This signal is part of the the Scenic Highway data transfer protocol between ViRGE/VX and the Scenic/MX2.
Video 8 in and Video 16 (PCI only) Modes		
LD[7:0]	I	LPB Data Bus [7:0]. This is the Scenic Highway data bus and carries video data input.
LD[15:8]	I	(PCI) LPB Data Bus [15:8]. Scenic Highway video data input for the upper data byte in Video 16 mode.
HS	I	HSYNC. HSYNC input signaling the transition from one line to the next.
VS	I	VSYNC. VSYNC input signaling the transition from one frame to the next.
TEST		
SCANEN	I	Used for S3 testing.
SCANOUT	O	Used for S3 testing.
TESTCLK	I	Used for S3 testing.
TESTEN	I	Used for S3 testing.
POWER AND GROUND		
VDDA	I	Analog power supply (non-clocks)
VDDACLK[1:2]	I	Analog power supply for clocks (CLK1 = MCLK, CLK2 = DCLK)
VDD	I	Digital power supply
VSSA	I	Analog ground
VSS	I	Digital ground

3.3 PIN LISTS

Table 3-2 lists all pins alphabetically. Table 3-3 lists all pins in numerical order.

Table 3-2. Alphabetical Pin Listing

Name	PIN(S)
AB	A18
AD[31:0]	D5, A4, C5, B3, C4, A3, B1, C2, D3, D1, E2, E4, E3, E1, F2, G4
	K2, J3, K1, K4, K3, M2, M1, L3, M4, N1, M3, P2, P4, P1, N3, R2
AG	A19
AR	A22
$\overline{\text{BLANK}}$	A8
$\overline{\text{CAS}}$[7:0]	AE16, AD15, AF16, AC15, AE19, AF19, AD18, AE20
$\overline{\text{C/BE}}$[3:0]	C1, F3, J2, N2
COMP	B19
$\overline{\text{CREQ}}$/CRDY	A5
DACD[7:0]	AD26, AC25, AC24, AC26, AB25, AB23, AB24, AB26
DACLCLK	AD23
$\overline{\text{DACRD}}$	AD22
DACSCLK	AD25
$\overline{\text{DACWR}}$	AE24
$\overline{\text{DEVSEL}}$	H2
DSF	AD14
EDCLK	C21
EMCLK	A16
$\overline{\text{ENFEAT}}$	A16
ESYNC	B10
EVCLK	A5
EVIDEO	B5
$\overline{\text{FRAME}}$	F1
$\overline{\text{GNT}}$	C8
GOP0	A16
GOP1	B8
HS	B5
HSYNC	B16
IDSEL	D2
$\overline{\text{INTA}}$	B15
$\overline{\text{IRDY}}$	G1
LCLK	D17
LD[15:0]	C7, A6, D7, A9, C10, B9, A10, C11, A13, D13, B13, A14, D15, B14, C16, A15
MA[8:0]	AE21, AF21, AD20, AE22, AF22, AD21, AE23, AC22, AF23

Table 3-2. Alphabetical Pin Listing (Continued)

Name	PIN(S)
N/C	A11, A12, A17, B11, B12, B18, C12, C13, C15, D10, D12, D20, F25, G2,
	H1, L1, N25, T2, T25, W4, AA24, AC3, AC20, AD1, AD17, AE10, AE13,
	AE26, AF15, AF24
$\overline{OE[1:0]}$	AD16, AE18
PA[15:8]	C7, A6, D7, A9, C10, B9, A10, C11, A13, D13, B13, A14, D15, B14, C16, A15
PAR	H3
PD[63:0]	AF3, AE4, AD4, AF4, AE5, AC5, AF5, AE6, AD6, AF6, AE7, AF7,
	AD7, AE8, AC9, AF8, AD8, AE9, AF9, AD9, AF10, AC10, AE11, AD10,
	AF11, AE12, AF12, AD11, AC12, AF13, AD12, AE14, AD26, AC25, AC24,
	AC26, AB25, AB23, AB24, AB26, Y23, AA26, Y25, Y26, Y24, W25, V23,
	W26, W24, V25, V26, U25, V24, U26, U23, U24, T26, R25, R26,
	T24, P25, R23, P26, R24
$\overline{PDOWN}$	C21
RA[15:0]	AD8, AE9, AF9, AD9, AF10, AC10, AE11, AD10, AF11, AE12, AF12,
	AD11, AC12, AF13, AD12, AE14
$\overline{RAS[1:0]}$	AE17, AF18
RD[7:0]	AD6, AF6, AE7, AF7, AD7, AE8, AC9, AF8
$\overline{REQ}$	B6
$\overline{RESET}$	C9
$\overline{ROMEN}$	D24
RS[3:0]	Y23, AA26, Y25, Y26
RSET	A20
SC[3:0]	C13, B12, AC14, AD19
SCANEN	B11
SCANOUT	H3
SCLK	D8
$\overline{SE[3:0]}$	AE15, AD13, AC19, AF20
SID[63:0]	P3, R1, R3, T1, R4, U2, T3, U1, U4, V2, U3, V1, W2, W1, V3, Y2, Y1, W3,
	AA2, Y4, AA1, Y3, AB2, AB1, AA3, AC2, AB4, AC1, AB3, AD2, AF2, AE3,
	M25, N24, M26, L25, M24, L26, M23, K25, L24, K26, K23, J25, K24, J26,
	H25, H26, J24, G25, H23, G26, H24, G23, F26, G24, E25, E26, F24, D25,
	E23, D26, E24, C25
SPCLK	C6
SPD	B4
$\overline{STOP}$	J4
$\overline{STWR}$	B8
TESTCLK	H1
TESTEN	C12
$\overline{TRDY}$	G3
VCLK	D17
VCLKI	A7

Table 3-2. Alphabetical Pin Listing (Continued)

Name	PIN(S)
VDDACLK1	A23
VDDACLK2	B23, B24
VDD	B20, B21, C14, C18, C19, D6, D11, D16, D21, D22, F4, F23, L2, L4, L23, N23, T4,
	T23, AA4, AA23, AC6, AC11, AC16, AC21, AF14
VREF	C20
$\overline{\text{VREQ}}$/VRDY	B5
VS	A5
VSS	A1, A2, A21, A24, A26, B2, B7, B17, B22, B25, B26, C3, C22, C23, C24, D4, D9,
	D14, D18, D19, D23, H4, J1, J23, L11, L12, L13, L14, L15, L16, M11, M12,
	M13, M14, M15, M16, N4, N11, N12, N13, N14, N15, N16, N26, P11, P12, P13,
	P14, P15, P16, P23, P24, R11, R12, R13, R14, R15, R16, T11, T12, T13, T14
	T15, T16, V4, W23, AA25, AC4, AC7, AC8, AC13, AC18, AC23, AD3, AD5, AD24,
	AE1, AE2, AE25, AF1, AF25, AF26
VSYNC	C17
$\overline{\text{WE}}$[1:0]	AF17, AC17
XIN	C26
XOUT	A25

Table 3-3. Numerical Pin Listing

Number	Name	Number	Name
A1	**VSS**	B18	N/C
A2	**VSS**	B19	COMP
A3	AD26	**B20**	**VDD**
A4	AD30	**B21**	**VDD**
A5	$\overline{\text{CREQ}}$/CRDY/VS/EVCLK	**B22**	**VSS**
A6	LD14/PA14	**B23**	**VDDACLK2**
A7	VCLKI	**B24**	**VDDACLK2**
A8	$\overline{\text{BLANK}}$	**B25**	**VSS**
A9	LD12/PA12	**B26**	**VSS**
A10	LD9/PA9	C1	$\overline{\text{C/BE3}}$
A11	N/C	C2	AD24
A12	N/C	**C3**	**VSS**
A13	LD7/PA7	C4	AD27
A14	LD4/PA4	C5	AD29
A15	LD0/PA0	C6	SPCLK
A16	$\overline{\text{ENFEAT}}$/GOP0/EMCLK	C7	LD15/PA15
A17	N/C	C8	$\overline{\text{GNT}}$
A18	AB	C9	$\overline{\text{RESET}}$
A19	AG	C10	LD11/PA11
A20	RSET	C11	LD8/PA8
A21	**VSS**	C12	TESTEN
A22	AR	C13	SC3
A23	**VDDACLK1**	**C14**	**VDD**
A24	**VSS**	C15	N/C
A25	XOUT	C16	LD1/PA1
A26	**VSS**	C17	VSYNC
B1	AD25	**C18**	**VDD**
B2	**VSS**	**C19**	**VDD**
B3	AD28	C20	VREF
B4	SPD	C21	$\overline{\text{PDOWN}}$/EDCLK
B5	$\overline{\text{VREQ}}$/VRDY/HS/EVIDEO	**C22**	**VSS**
B6	$\overline{\text{REQ}}$	**C23**	**VSS**
B7	**VSS**	**C24**	**VSS**
B8	$\overline{\text{STWR}}$/GOP1	C25	SID0
B9	LD10/PA10	C26	XIN
B10	ESYNC	D1	AD22
B11	SCANEN	D2	IDSEL
B12	SC2	D3	AD23
B13	LD5/PA5	**D4**	**VSS**
B14	LD2/PA2	D5	AD31
B15	$\overline{\text{INTA}}$	**D6**	**VDD**
B16	HSYNC	D7	LD13/PA13
B17	**VSS**	D8	SCLK

Table 3-3. Numerical Pin Listing (Continued)

Number	Name	Number	Name
D9	**VSS**	H2	$\overline{\text{DEVSEL}}$
D10	N/C	H3	PAR/SCANOUT
D11	**VDD**	**H4**	**VSS**
D12	N/C	H23	SID13
D13	LD6/PA6	H24	SID11
D14	**VSS**	H25	SID17
D15	LD3/PA3	H26	SID16
D16	**VDD**	**J1**	**VSS**
D17	LCLK/VCLK	J2	$\overline{\text{C/BE1}}$
D18	**VSS**	J3	AD14
D19	**VSS**	J4	$\overline{\text{STOP}}$
D20	N/C	**J23**	**VSS**
D21	**VDD**	J24	SID15
D22	**VDD**	J25	SID20
D23	**VSS**	J26	SID18
D24	$\overline{\text{ROMEN}}$	K1	AD13
D25	SID4	K2	AD15
D26	SID2	K3	AD11
E1	AD18'	K4	AD12
E2	AD21	K23	SID21
E3	AD19	K24	SID19
E4	AD20	K25	SID24
E23	SID3	K26	SID22
E24	SID1	L1	N/C
E25	SID7	**L2**	**VDD**
E26	SID6	L3	AD8
F1	$\overline{\text{FRAME}}$	**L4**	**VDD**
F2	AD17	**L11**	**VSS**
F3	$\overline{\text{C/BE2}}$	**L12**	**VSS**
F4	**VDD**	**L13**	**VSS**
F23	**VDD**	**L14**	**VSS**
F24	SID5	**L15**	**VSS**
F25	N/C	**L16**	**VSS**
F26	SID9	**L23**	**VDD**
G1	$\overline{\text{IRDY}}$	L24	SID23
G2	N/C	L25	SID28
G3	$\overline{\text{TRDY}}$	L26	SID26
G4	AD16	M1	AD9
G23	SID10	M2	AD10
G24	SID8	M3	AD5
G25	SID14	M4	AD7
G26	SID12	**M11**	**VSS**
H1	TESTCLK	**M12**	**VSS**

Table 3-3. Numerical Pin Listing (Continued)

Number	Name	Number	Name
M13	**VSS**	**R14**	**VSS**
M14	**VSS**	**R15**	**VSS**
M15	**VSS**	**R16**	**VSS**
M16	**VSS**	R23	PD2
M23	SID25	R24	PD0
M24	SID27	R25	PD6
M25	SID31	R26	PD5
M26	SID29	T1	SID60
N1	AD6	T2	N/C
N2	$\overline{C/BE0}$	T3	SID57
N3	AD1	**T4**	**VDD**
N4	**VSS**	**T11**	**VSS**
N11	**VSS**	**T12**	**VSS**
N12	**VSS**	**T13**	**VSS**
N13	**VSS**	**T14**	**VSS**
N14	**VSS**	**T15**	**VSS**
N15	**VSS**	**T16**	**VSS**
N16	**VSS**	**T23**	**VDD**
N23	**VDD**	T24	PD4
N24	SID30	T25	N/C
N25	N/C	T26	PD7
N26	**VSS**	U1	SID56
P1	AD2	U2	SID58
P2	AD4	U3	SID53
P3	SID63	U4	SID55
P4	AD3	U23	PD9
P11	**VSS**	U24	PD8
P12	**VSS**	U25	PD12
P13	**VSS**	U26	PD10
P14	**VSS**	V1	SID52
P15	**VSS**	V2	SID54
P16	**VSS**	V3	SID49
P23	**VSS**	**V4**	**VSS**
P24	**VSS**	V23	PD17
P25	PD3	V24	PD11
P26	PD1	V25	PD14
R1	SID62	V26	PD13
R2	AD0	W1	SID50
R3	SID61	W2	SID51
R4	SID59	W3	SID46
R11	**VSS**	W4	N/C
R12	**VSS**	**W23**	**VSS**
R13	**VSS**	W24	PD15

Table 3-3. Numerical Pin Listing (Continued)

Number	Name	Number	Name
W25	PD18	**AC18**	**VSS**
W26	PD16	AC19	$\overline{SE1}$
Y1	SID47	AC20	N/C
Y2	SID48	**AC21**	**VDD**
Y3	SID42	AC22	MA1
Y4	SID44	**AC23**	**VSS**
Y23	PD23/RS3	AC24	PD29/DACD5
Y24	PD19	AC25	PD30/DACD6
Y25	PD21/RS1	AC26	PD28/DACD4
Y26	PD20/RS0	AD1	N/C
AA1	SID43	AD2	SID34
AA2	SID45	**AD3**	**VSS**
AA3	SID39	AD4	PD61
AA4	**VDD**	**AD5**	**VSS**
AA23	**VDD**	AD6	PD55/RD7
AA24	N/C	AD7	PD51/RD3
AA25	**VSS**	AD8	PD47/RA15
AA26	PD22/RS2	AD9	PD44/RA12
AB1	SID40	AD10	PD40/RA8
AB2	SID41	AD11	PD36/RA4
AB3	SID35	AD12	PD33/RA1
AB4	SID37	AD13	$\overline{SE2}$
AB23	PD26/DACD2	AD14	DSF
AB24	PD25/DACD1	AD15	$\overline{CAS6}$
AB25	PD27/DACD3	AD16	$\overline{OE1}$
AB26	PD24/DACD0	AD17	N/C
AC1	SID36	AD18	$\overline{CAS1}$
AC2	SID38	AD19	SC0
AC3	N/C	AD20	MA6
AC4	**VSS**	AD21	MA3
AC5	PD58	AD22	$\overline{DACRD}$
AC6	**VDD**	AD23	DACLCLK
AC7	**VSS**	**AD24**	**VSS**
AC8	**VSS**	AD25	DACSCLK
AC9	PD49/RD1	AD26	PD31/DACD7
AC10	PD42/RA10	**AE1**	**VSS**
AC11	**VDD**	**AE2**	**VSS**
AC12	PD35/RA3	AE3	SID32
AC13	**VSS**	AE4	PD62
AC14	SC1	AE5	PD59
AC15	$\overline{CAS4}$	AE6	PD56
AC16	**VDD**	AE7	PD53/RD5
AC17	$\overline{WE0}$	AE8	PD50/RD2

Table 3-3. Numerical Pin Listing (Continued)

Number	Name	Number	Name
AE9	PD46/RA14	AF5	PD57
AE10	N/C	AF6	PD54/RD6
AE11	PD41/RA9	AF7	PD52/RD4
AE12	PD38/RA6	AF8	PD48/RD0
AE13	N/C	AF9	PD45/RA13
AE14	PD32/RA0	AF10	PD43/RA11
AE15	$\overline{SE3}$	AF11	PD39/RA7
AE16	$\overline{CAS7}$	AF12	PD37/RA5
AE17	$\overline{RAS1}$	AF13	PD34/RA2
AE18	$\overline{OE0}$	**AF14**	**VDD**
AE19	$\overline{CAS3}$	AF15	N/C
AE20	$\overline{CAS0}$	AF16	$\overline{CAS5}$
AE21	MA8	AF17	$\overline{WE1}$
AE22	MA5	AF18	$\overline{RAS0}$
AE23	MA2	AF19	$\overline{CAS2}$
AE24	$\overline{DACWR}$	AF20	$\overline{SE0}$
AE25	**VSS**	AF21	MA7
AE26	N/C	AF22	MA4
AF1	**VSS**	AF23	MA0
AF2	SID33	AF24	N/C
AF3	PD63	**AF25**	**VSS**
AF4	PD60	**AF26**	**VSSB**

Section 4: Electrical Data

4.1 MAXIMUM RATINGS

Table 4-1. Absolute Maximum Ratings

Ambient temperature	0° C to 70° C
Storage temperature	-40° C to 125° C
DC Supply Voltage	-0.5V to 7.0V
I/O Pin Voltage with respect to V_{SS}	-0.5V to V_{DD}+0.5V

4.2 DC SPECIFICATIONS

Note: In all cases below, digital VDD = 5V ± 5% and the operating temperature is 0° C to 70° C.

Table 4-2. RAMDAC/Clock Synthesizer DC Specifications

Symbol	Parameter	Min	Typical	Max	Unit
AVDD	DAC supply voltage	4.75	5	5.25	V
AVDD (CLOCK)	PLL supply voltage	4.75	5	5.25	V
VREF	Internal voltage reference	1.10	1.235	1.35	V

Table 4-3. RAMDAC Characteristics

	Min	Typical	Max	Unit
Resolution Each DAC		8		bits
LSB Size		66		µA
Integral Linearity Error			± 1	LSB
Differential Linearity Error			± 1	LSB
Output Full-Scale Current	15.4	17.6	19.8	mA
DAC to DAC Mismatch			5%	
Power Supply Rejection Ratio			0.5	%/ % AVDD
Output Compliance	0.0		1.5	V
Output Capacitance			30	pF
Glitch Impulse		75		pV-Sec

Table 4-4. Digital DC Specifications (VDD = 5V ± 5%, Operating Temperature 0° C to 70° C)

Symbol	Parameter	Min	Max	Unit
V_{IL}	Input Low Voltage	-0.5	0.8	V
V_{IH}	Input High Voltage	2.4	V_{DD} +0.5	V
V_{OL}	Output Low Voltage		V_{SS} + 0.4	V
V_{OH}	Output High Voltage	2.4		V
I_{OL1}	Output Low Current	8 (Note 1)		mA
I_{OH1}	Output High Current	-4		mA
I_{OL2}	Output Low Current	16 (Note 2)		mA
I_{OH2}	Output High Current	-8		mA
I_{OL3}	Output Low Current	24 (Note 3)		mA
I_{OH3}	Output High Current	-12		mA
I_{OZ}	Output Tri-state Current		1	µA
C_{IN}	Input Capacitance		5	pF
C_{OUT}	Output Capacitance		5	pF
I_{CC}	Power Supply Current		740 (Note 4)	mA

Notes for Table 4-4

1. I_{OL1}, I_{OH1} for pins $\overline{\text{ROMEN}}$, $\overline{\text{INTA}}$, $\overline{\text{STRD}}$, $\overline{\text{STWR}}$, HSYNC, VSYNC, VCLK, $\overline{\text{BLANK}}$, $\overline{\text{ENFEAT}}$, $\overline{\text{CAS[7:0]}}$, PD[63:0], AD[31:0], LD[7:0], $\overline{\text{VREQ}}$/VRDY, SPCLK, $\overline{\text{DACRD}}$, $\overline{\text{DACWR}}$, $\overline{\text{REQ}}$ SPD. The I_{OL} for $\overline{\text{CAS[7:0]}}$ and/or PD[63:0] can be changed to 4 mA via CR80.
2. I_{OL2}, I_{OH2} for pins $\overline{\text{OE[1:0]}}$, $\overline{\text{WE[1:0]}}$, $\overline{\text{RAS[1:0]}}$, MA[8:0], SC[3:0], $\overline{\text{SE[3:0]}}$, DSF[2:0]. The I_{OL} for any of these signals can be changed to 8 mA via CR80.
3. I_{OL3}, I_{OH3} for pins PAR, $\overline{\text{STOP}}$, $\overline{\text{DEVSEL}}$, $\overline{\text{TRDY}}$
4. Maximum current that can be dissipated based on operation at 70° C and 5.25V and a maximum junction temperature of 125° C .

4.3 AC SPECIFICATIONS

4.3.1 RAMDAC AC Specifications

Table 4-5. RAMDAC AC Specifications

Parameter	Typical	Max	Unit	Notes
DAC Output Delay	5		ns	1
DAC Output Rise/Fall Time	3		ns	2
DAC Output Settling Time	15		ns	
DAC-to-DAC Output Skew	2	5	ns	3

Notes for Table 4-5

1. Measured from the 50% point of ICLK to the 50% point of full scale transition
2. Measured from 10% to 90% full scale
3 With DAC outputs equally loaded

Table 4-6. RAMDAC Output Specifications

Description	I_{OUT} (mA)	V_{OUT} (V)	BLANK	Input Data
White (with $\overline{BLANK}$ pedestal)	19.05 (typical)	0.714 (typical)	1	FFH
White	17.6 (typical)	0.66 (typical	1	FFH
Data (with $\overline{BLANK}$ pedestal)	Data + 1.45	Data + 0.054	1	Data
Data	Data	Data	1	Data
Black (with $\overline{BLANK}$ pedestal)	1.45 (typical)	0.054	1	00H
Black (no pedestal)	0	0	1	00H
$\overline{BLANK}$	0	0	0	Don't Care

Notes for Table 4-6

1. Condition for V_{OUT} is a 75 Ohm doubly terminated load, use of the internal VREF and RSET = 147 Ohms.
2, No sync pedestal is provided. Sync output levels are the same as for black output.
3. The $\overline{BLANK}$ pedestal is active when SR1A_5 = 1.

4.3.2 Clock Timing

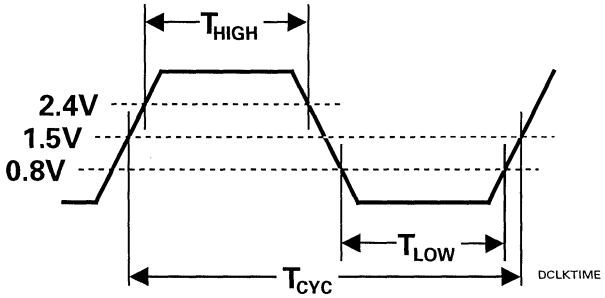


Figure 4-1. Clock Waveform Timing

Table 4-7. Clock Waveform Timing

Symbol	Parameter	Min	Max	Units	Notes
T_{CYC}	SCLK Cycle Time	30	125	ns	1
	LCLK Cycle Time	30	200	ns	
	MCLK Cycle Time	20	100	ns	
	DCLK Cycle Time (VGA Mode)	25	100	ns	1
	DCLK Cycle Time (Enhanced Mode)	12.5	100	ns	1
	SC Cycle Time	18.2	100	ns	
T_{HIGH}	SCLK High Time	12	80	ns	
	LCLK High Time	12	160	ns	
	SC High Time	10	80	ns	
T_{LOW}	SCLK Low Time	12	80	ns	
	LCLK Low Time	12	160	ns	
	SC Low Time	10	80	ns	
	SCLK, LCLK, SC Slew Rate	1	4	V/ns	2

Notes

1. $f_{DCLK} \geq 1/2\ f_{SCLK}$ to ensure valid writes to the PLLs.
2. Rise and fall times are specified in terms of the edge rate measured in V/ns. This slew rate must be met across the minimum peak-to peak portion of the clock waveform.

4.3.3 Input/Output Timing

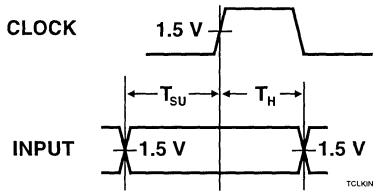


Figure 4-2. Input Timing

Table 4-8. SCLK-Referenced Input Timing

PCI Bus			
Symbol	**Parameter**	**Min**	**Units**
T_{SU}	AD[31:0], C/BE[3:0], FRAME, IRDY, IDSEL setup	7	ns
T_H	AD[31:0] hold	1	ns
T_H	C/BE[3:0], FRAME, IRDY, IDSEL hold	1	ns
T_{SU}	GNT setup	10	ns
T_H	GNT hold	0	ns
Miscellaneous			
Symbol	**Parameter**	**Min**	**Units**
T_{SU}	RD[7:0] setup	5	ns
T_H	RD[7:0] hold	7	ns

Table 4-9. LCLK-Referenced Input Timing

Scenic/MX2 Interface			
Symbol	**Parameter**	**Min**	**Units**
T_{SU}	LD[7:0] setup	10	ns
T_H	LD[7:0] hold	9	ns
T_{SU}	CREQ/CRDY	6	ns
T_H	CREQ/CRDY	8	ns
SAA7110 Interface			
Symbol	**Parameter**	**Min**	**Units**
T_{SU}	LD[7:0] setup (also LD[15:8] for 16-bit interface)	6	ns
T_H	LD[7:0] hold (also LD[15:8] for 16-bit interface)	8	ns
T_{SU}	HS setup	6	ns
T_H	HS hold	7	ns
T_{SU}	VS setup	6	ns
T_H	VS hold	7	ns

Table 4-10. MCLK-Referenced Input Timing

Symbol	Parameter	Min	Units
T_{SU}	MD[63:0] setup to MCLK high (2-cycle EDO)	0	ns
T_H	MD[63:0] hold from MCLK high (2-cycle EDO)	12.5	ns
T_{SU}	MD[63:0] setup to following $\overline{CAS}$ low (1-cycle EDO)	0	ns
T_H	MD[63:0] hold from following $\overline{CAS}$ low (1-cycle EDO)	15	ns
T_{SU}	MD[63:0] setup to $\overline{CAS}$ high (fast page)	0	ns
T_H	MD[63:0] hold from $\overline{CAS}$ high (fast page)	15	ns

Note

1. The timing reference in each of the three cases above is to the event that causes the latching of the read data. The MCLK used to latch 2-cycle EDO data is an internal signal that cannot be directly observed. The $\overline{CAS}$ signals used to latch read data in the other operational modes are derived from the internal MCLK.

Table 4-11. SC-Referenced Input Timing

Symbol	Parameter	Min	Units
T_{SU}	SID[63:0] setup	6	ns
T_H	SID[63:0] hold	0	ns

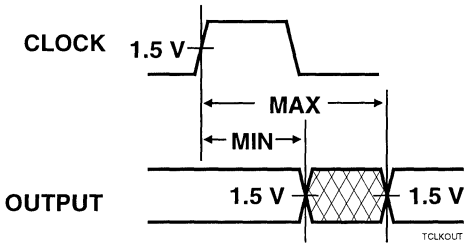


Figure 4-3. Output Timing

The minimum delay is the minimum time after the clock edge that the valid signal state from the previous cycle will begin transition to the next state (become invalid).

The maximum delay is the maximum time after the clock edge that the signal state is valid for the next cycle.

Table 4-12. SCLK-Referenced Output Timing

PCI Bus					
Parameter	**Min**	**Max**	**Units**	**Load(pF)**	**Notes**
AD[31:0] valid delay	2	16	ns	50	1
$\overline{\text{DEVSEL}}$, PAR delay	2	11	ns	30	Medium $\overline{\text{DEVSEL}}$ timing used
$\overline{\text{STOP}}$ delay	2	11	ns	30	
$\overline{\text{TRDY}}$ delay	2	11	ns	50	
$\overline{\text{INTA}}$ delay	2	11	ns	50	
$\overline{\text{REQ}}$ delay	2	10	ns		
Miscellaneous					
$\overline{\text{ROMEN}}$ delay	4	10	ns	50	
ROM address valid delay	5	30	ns	30	
AD[7:0] ROM data valid delay	5	30	ns	50	
$\overline{\text{DACRD}}$ delay	2	11	ns	20	
$\overline{\text{DACWR}}$ delay	2	11	ns	20	
RS[3:0] delay	2	11	ns	20	
RA[15:0] valid delay	2	11	ns	30	

Note

1. Due to the timing for $\overline{\text{TRDY}}$ for read cycles, data is not sampled on the clock edge immediately following its becoming valid. This guarantees the PCI 2.1 specification time of 11 ns.

Table 4-13. LCLK-Referenced Output Timing

Scenic/MX2 Interface					
Parameter	**Min**	**Max**	**Units**	**Load(pF)**	**Notes**
$\overline{\text{VREQ}}$/VRDY active delay	2	11	ns	20	7 ns typ
LD[7:0] valid delay	2	15	ns	20	8 ns typ
LD[7:0] tri-state from LCLK	7	15	ns	20	

Table 4-14. MCLK-Referenced Output Timing

Parameter	**Min**	**Max**	**Units**	**Load(pF)**	**Notes**
PD[63:0] valid delay	2	7/11	ns	70	1
MA[8:0] valid delay	1.5	8	ns	65	
$\overline{\text{CAS}}$[7:0] active delay	1	5.5	ns	30	
$\overline{\text{CAS}}$[7:0] inactive delay	1	5.5	ns	30	
$\overline{\text{RAS}}$[1:0] active delay	1	5	ns	80	
$\overline{\text{RAS}}$[1:0] inactive delay	1	6.5	ns	80	
$\overline{\text{OE}}$[1:0] active delay	1.5	4.5	ns	50	
$\overline{\text{WE}}$[1:0] active delay	1.5	4.5	ns	50	
DSF[2:0] delay	2	10	ns	80	2

Notes

1. The maximum delay time is 7 ns for 1-cycle operation and 11 ns for 2-cycle operation.
2. DSF timing is based on the falling edge of MCLK.
3. MCLK is an internal clock

Table 4-15. ICLK-Referenced Output Timing

Parameter	**Min**	**Max**	**Units**	**Load(pF)**	**Notes**
$\overline{\text{SE}}$[3:0] valid before SC[1:0] rising edge	4		ns	80	1
$\overline{\text{BLANK}}$ delay	3	6	ns	30	2
HSYNC delay	1	3	ns	50	2
VSYNC delay	1	3	ns	30	2

Note

1. SC is derived from ICLK.
2. ICLK is an internal clock.

Table 4-16. Feature Connector Timing - Output from ViRGE/VX to Feature Connector

Symbol	Parameter	Min	Units	Notes
T_{SU}	PA[15:0], $\overline{BLANK}$ setup to VCLK rising	5	ns	
T_H	PA[15:0], $\overline{BLANK}$ hold from VCLK rising	5	ns	

Table 4-17. Feature Connector Timing - Output from Feature Connector to ViRGE/VX

Symbol	Parameter	Min	Max	Units	Notes
T_{SU}	PA[15:0], $\overline{BLANK}$ setup to VCLK or VCLKI rising	6		ns	1
T_H	PA[15:0], $\overline{BLANK}$ hold from VCLK or VCLKI rising	6		ns	1
	VCLK	25	40	ns	1
	VCLKI	27	40	ns	1, 2
	VCLK, VCLKI duty cycle	40	60	%	
	VCLK, VCLKI high time	10	25	ns	
	VCLK, VCLKI low time	10	25	ns	
	VCLK, VCLKI slew rate	1	4	V/ns	

Notes

1. Pixel data is clocked into the internal RAMDAC using VCLK for a pass-through feature connector and VCLKI for a VAFC configuration.
2. This corresponds to the VESA VAFC specification of a maximum clock of 37.5 MHz.

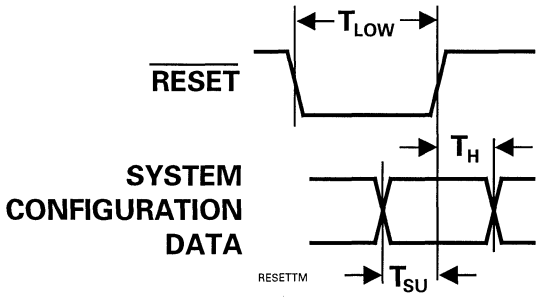


Figure 4-4. Reset Timing

Table 4-18. Reset Timing

Symbol	Parameter	Min	Units
T_{LOW}	$\overline{\text{RESET}}$ active pulse width	1000	ns
T_{SU}	PD[28:0] setup to $\overline{\text{RESET}}$ inactive	20	ns
T_H	PD[28:0] hold from $\overline{\text{RESET}}$ inactive	10	ns

Section 5: Reset and Initialization

The reset signal ($\overline{\text{RESET}}$) resets the internal state machines in ViRGE/VX and places all registers in their power-on default states. It also initiates several configuration actions, as described in this section.

The PD[28:0] pins are pulled up internally . They can be individually pulled low through external 10 KΩ resistors. These pull-ups and pull-downs do not affect normal operation of the pins as part of the pixel data bus, but they do force the pins to a definite state during reset. At the rising edge of the reset signal, this state is sampled and the data loaded into the CR36, CR37, CR68 and CR6F registers. The data is used for system configuration, such as system bus and memory parameter selection. The definitions of the PD[28:0] strapping bits at the rising edge of the reset signal are shown in Table 5-1.

Strapping bits 6-5 define the display memory size. However, the S3 BIOS determines this value directly and writes it to CR36_6-5 after reset. Therefore, systems using the S3 BIOS do not need to strap the PD[6:5] pins. Other pins may also not require strapping, depending on the design and bus type.

Table 5-1. Definition of PD[28:0] at the Rising Edge of the Reset Signal

CR Bits	PD Bits	Value	Function
CR36_1-0	1-0		Reserved
	Memory Page Mode Select		
CR36_3-2	3-2	00	VRAM 1-cycle EDO mode
		01	Reserved
		10	VRAM 2-cycle EDO) mode
		11	VRAM fast page mode
CR36_4	4		Reserved
	Display Memory Size		
CR36_6-5	6-5	00	2 MBytes
		01	4 MBytes
		10	6 MBytes
		11	8 MBytes
	8-Column Block Write Support		
CR36_7	7	0	VRAM does not support 8-column block writes
		1	VRAM supports 8-column block writes
CR37_0	8		Reserved
	Tri-state Outputs		
CR37_1		0	All outputs tri-stated and all bi-directional pins become inputs
		1	Normal Operation
CR37_2	10		Reserved
	Test Mode		
CR37_3	11	0	Test mode. Use external DCLK, MCLK (test purposes only)
		1	Use internal DCLK, MCLK
CR37_4	12		Reserved
	DRAM Size		
CR37_6-5	14-13	00	Reserved
		01	4 MBytes DRAM
		10	2 MBytes DRAM
		11	0 MByte DRAM
CR37_7	15		Reserved

Table 5-1. Definition of PD[28:0] at the Rising Edge of the Reset Signal (Continued)

CR Bits	PD Bits	Value	Function
	$\overline{CAS}$/$\overline{OE}$ Adjust		
CR68_1-0	17-16	00	approximately 6.5 ns adjustment
		01	approximately 5 ns adjustment
		10	approximately 3.5 ns adjustment
		11	no adjustment
	$\overline{RAS}$ Low Timing Select		
CR68_2	18	0	4.5 MCLKs
		1	3.5 MCLKs
	$\overline{RAS}$ Pre-Charge Timing Select		
CR68_3	19	0	3.5 MCLKs
		1	2.5 MCLKs
CR68_7-4	23-20		Reserved
CR6F_0	24	0	Reserved
	Serial Port I/O Address Select		
CR6F_1	25	0	Serial Port register accessed at I/O address 000E8H
		1	Serial Port register accessed at I/O address 000E2H
	Serial Port Address Type Select		
CR6F_2	26	0	Serial Port register accessed at address defined in CR6F_1
		1	Serial Port register accessed at its MMIO address only (offset FF20H)
	$\overline{WE}$ Delay		
CR6F_4-3	28-27	00	3 units delay
		01	2 units delay
		10	1 unit delay
		11	no delay

Section 6: PCI Bus Interface

6.1 OVERVIEW

ViRGE/VX provides a complete PCI interface. The pinout and other specifications are in conformance with Revision 2.1 of the the PCI specification. No glue logic is required.

6.2 PCI CONFIGURATION

The Vendor ID register (Index 00H) in the PCI Configuration space is hardwired to 5333H to specify S3 Incorporated as the vendor. The Device ID register is hardwired to 883DH. The Revision ID will vary by stepping.

Bits 10-9 of the Status register (Index 06H) are hardwired to 01b to specify medium DEVSEL timing. The Class Code register (Index 08H) is hardwired to 30000xxH to specify that ViRGE/VX is a VGA compatible device. Bits 3-0 of the Base Address 0 register (Index 10H) are hardwired to 00H. This indicates that the "prefetchable" bit is cleared to 0, the base register can be located anywhere in a 32-bit address space and the base register is located in memory space.

6.3 PCI CONTROLS

ViRGE/VX provides several methods of controlling PCI Bus operation. PCI disconnects are enabled via CR66_3 and CR66_7. The RAMDAC snoop method is selected via CR34_0. PCI master abort handling during RAMDAC snooping can be disabled via CR34_1. PCI retry handling during RAMDAC snooping can be disabled via CR34_2. PCI read burst cycles can be disabled via CR3A_7.

Section 7: Display Memory

ViRGE/VX supports a VRAM-based video frame buffer or a mixed VRAM and DRAM frame buffer The frame buffer can be 2-, 4-, 6- or 8 MBytes total. The size is specified via CR36_6-5. In a mixed VRAM/DRAM configuration, 2 or 4 MBytes of DRAM can be used for off-screen memory, such as to support 3D Z-buffering. The amount of DRAM installed is specified via CR37_6-5.

This section describes the various configurations supported, the functional timing for memory accesses and the operation of various register bits that affect memory timing and operation. It also describes how access to display memory is controlled to maximize graphics and video performance.

7.1 DISPLAY MEMORY CONFIGURATIONS

ViRGE/VX uses either fast page mode or extended data out (EDO) VRAMs or DRAMs for its frame buffer. All VRAMs/DRAMs can be configured as 256Kx8 or 256Kx16. The type of RAM operation is specified in CR36_3-2.

For loading reasons, a maximum of 16 RAM chips can be used for the frame buffer. Table 7-1 shows the supported memory size/chip count configurations. Note that if the signal drive levels are lowered via CR80_7-0, the maximum chip count will also be lowered.

Table 7-1 Memory Size/Chip Count Configurations (Maximum Drive)

	256Kx8	256Kx16
2 MB	8	4
4 MB	16	8
8 MB		16

Figure 7-1 shows a 4-MByte VRAM memory configuration using 256Kx16 VRAMs. This is a 64-bit serial SID bus configuration, selected by clearing CR66_5-4 to 00b. A 2-MByte configuration uses only the leftmost 4 chips in Figure 7-1. With 2- or 4-MByte configurations, either 2- or 4-MBytes of DRAM off screen memory are allowed. A 6-MByte configuration is also allowed, with the upper 2 MBytes available for off-screen memory.

An 8-MByte configuration duplicates the 4-MByte configuration except that $\overline{RAS1}$ is used to select the upper 4 MBytes. All configurations related to Figure 7-1 use the internal RAMDAC.

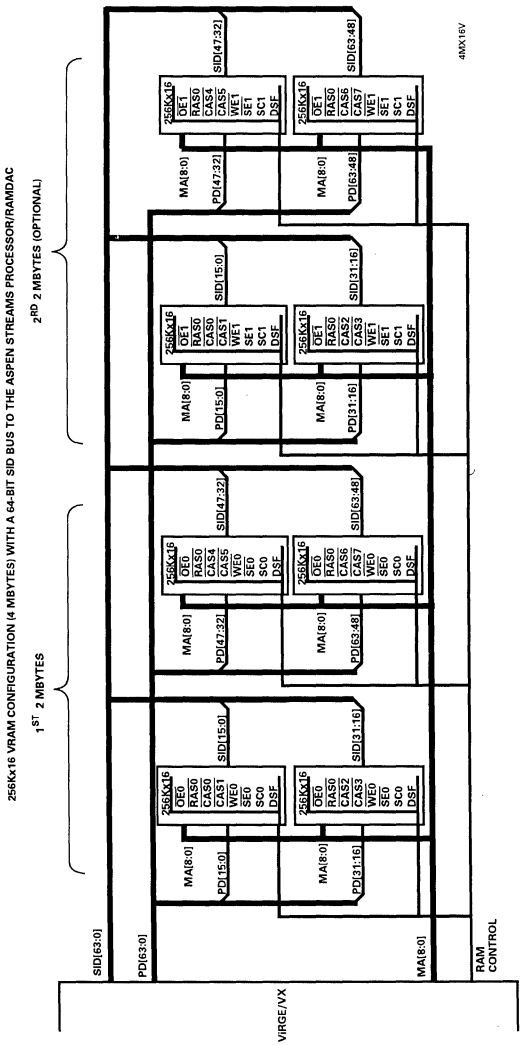


Figure 7-1. Internal RAMDAC Memory Configuration

Figure 7-2 shows an 8-MByte VRAM configuration for use with an external 128-bit RAMDAC. This configuration can support two modes of operation. The 128-bit parallel mode, selected when CR66_5-4 = 10b, uses two 64-bit SID busses to feed the 128-bit external RAMDAC. This provides the highest possible resolution, but does not make use of the ViRGE/VX Streams Processor.

A second mode of operation is available with the 8-MByte configuration by programming CR66_5-4 to 01b. This uses only 4 MBytes of the memory and a single 64-bit SID bus. The pixel data is fed back to ViRGE/VX and its Streams Processor. The top 8 chips in the memory configuration shown in Figure 7-2 are used for the 4 MBytes. ViRGE/VX automatically reconfigures the memory control signals to accomplish this. The lower 8 chips in Figure 7-2 cannot be accessed in this mode.

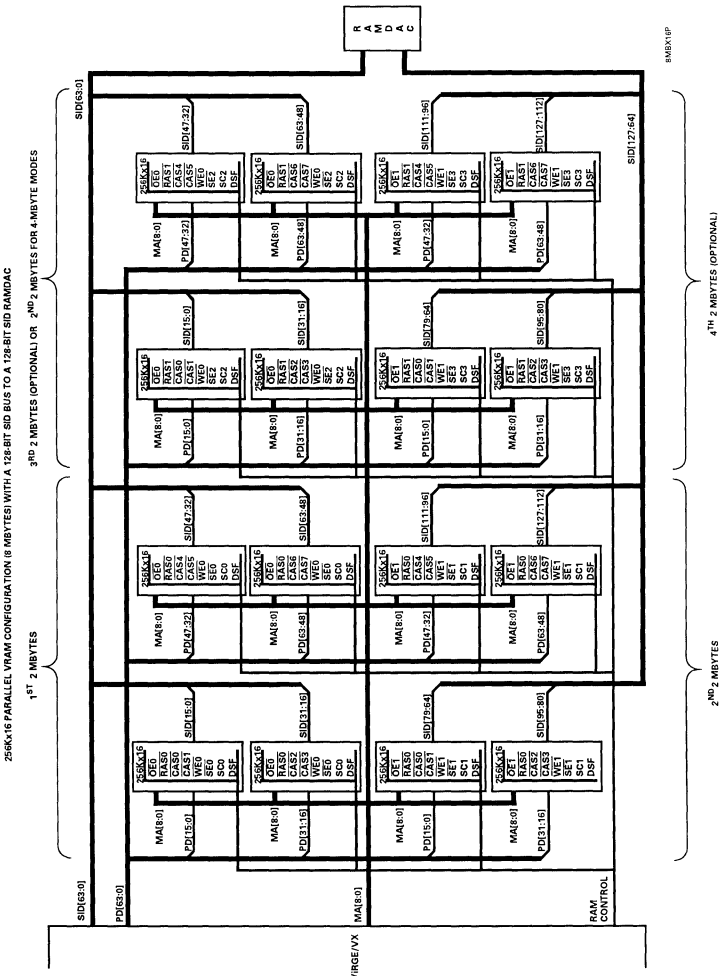


Figure 7-2. External RAMDAC Memory Configuration

This dual-mode configuration allows all 4-MByte graphics modes (all but one), to make use of the full range of ViRGE/VX features while also providing support for the high-end 1600x1200x24 8-MByte mode.

7.2 DISPLAY MEMORY REFRESH

ViRGE/VX uses the standard $\overline{CAS}$ before $\overline{RAS}$ refresh method for VRAMs and DRAMs. The functional timing for this can be found in the appropriate data book.

The number of refresh cycles performed per horizontal line is determined by bit 6 of CR11. If bit 2 of CR3A is set to 1, the number of refresh cycles per horizontal line is determined by the setting of bits 1-0 of CR3A.

7.3 FAST PAGE AND 2-CYCLE EDO FUNCTIONAL TIMING

Figure 7-3 shows the functional timing for a fast page mode read cycle. This also shows how certain parameters for various control signals can be adjusted to meet the access time requirements of a variety of DRAMs. Power-on strapping of CR68_0 allows the trailing edges of the $\overline{CAS}$ and $\overline{OE}$ signals to be delayed by 0 or 1 unit. (This unit, typically on the order of 1 to 2 ns, varies by signal loading, manufacturing process and other variables.) After power-up, MM8204_5 can be programmed to change the delay to 1 or 3 units. Power-on strapping of CR68_1 allows the leading edges of the $\overline{CAS}$ and $\overline{OE}$ signals to be delayed by 0 or 1 unit. After power-up, MM8204_6 can be programmed to change the delay to 1 or 3 units. Power-on strapping of CR68_2 allows selection of the $\overline{RAS}$ low time as 3.5 or 4.5 MCLKs. After power-up, MM8204_2 can be programmed to change this to 2.5 MCLKs. The low time can be increased 0.5 MCLK via CR58_7. Power-on strapping of CR68_3 allows selection of the $\overline{RAS}$ precharge time as 2.5 or 3.5 MCLKs. After power-up, MM8204_1 can be programmed to change this to 1.5 MCLKs. The high time can be reduced 0.5 MCLK via CR58_7.

Read data is latched on the rising edge of $\overline{CAS}$. An internal $\overline{CAS}$ is used for this purpose. The internal $\overline{OE}$ signal rises at the same time, but because of propagation delays, the DRAM will not see this edge immediately. This plus the DRAM turn-off time guarantees that valid data is latched.

Figure 7-4 shows the functional timing for a fast page mode write cycle. The $\overline{RAS}$ and $\overline{CAS}$ signals can be adjusted as explained for the read cycle above. Power-on strapping of CR6F_3 allows the trailing edge of the $\overline{WE}$ signal to be delayed by 0 or 1 unit. After power-up, MM8204_3 can be programmed to change the delay to 1 or 3 units. Power-on strapping of CR6F_4 allows the leading edge of the $\overline{WE}$ signal to be delayed by 0 or 1 unit. After power-up, MM8204_4 can be programmed to change the delay to 1 or 3 units.

Figure 7-5 shows the functional timing for a fast page mode read/modify/write cycle. This is a 1-wait state cycle.

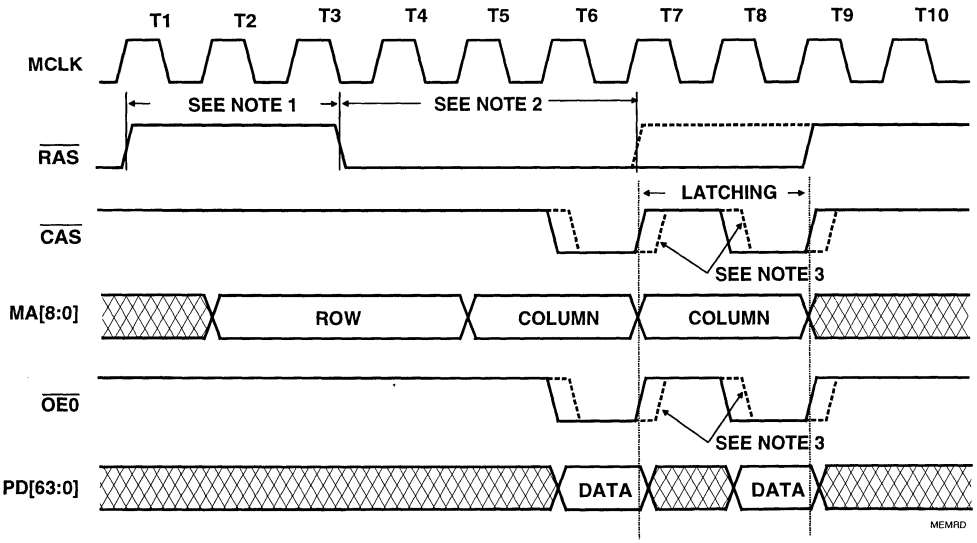


Figure 7-3. Fast Page Mode Read Cycle

Notes

1. The $\overline{RAS}$ precharge time can be adjusted via CR68_3, MM8204_1 and CR58_7.
2. The $\overline{RAS}$ low time for a single column access is adjustable via CR68_2, MM8204_2 and CR58_7. (The dashed line shows the $\overline{RAS}$ signal if the second page mode cycle were to be eliminated.)
3. The $\overline{CAS}$ and $\overline{OE}$ edges can be stretched via CR68_1-0 and MM8204_6-5. $\overline{CAS}$ and $\overline{OE}$ edges move together, but the leading and trailing edges can be stretched independently.

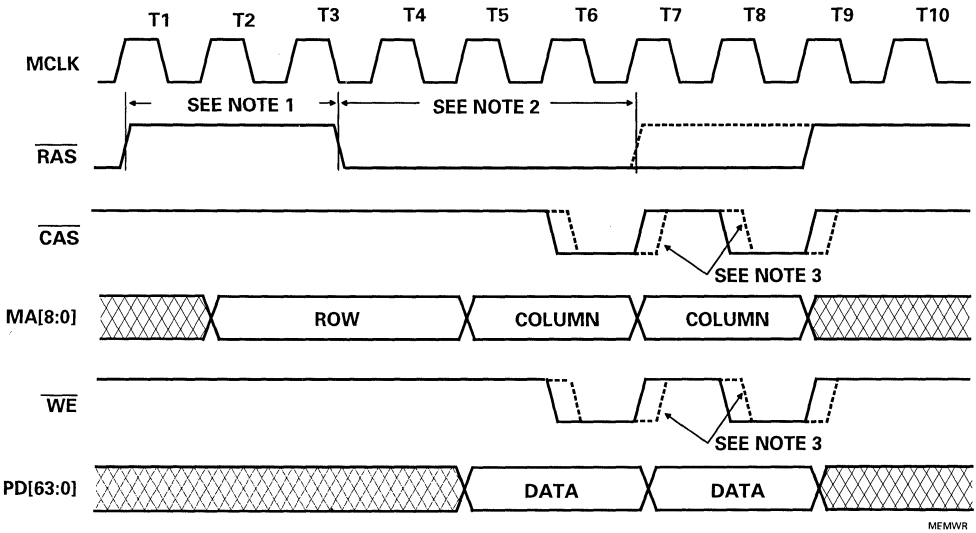


Figure 7-4. Fast Page Mode Write Cycle

Notes

1. The $\overline{RAS}$ precharge time can be adjusted via CR68_3, MM8204_1 and CR58_7.
2. The $\overline{RAS}$ low time for a single column access is adjustable via CR68_2, MM8204_2 and CR58_7. (The dashed line shows the $\overline{RAS}$ signal if the second page mode cycle were to be eliminated.)
3. The leading and trailing edges of $\overline{CAS}$ can be independently stretched via CR68_1-0 and MM8204_6-5. The leading and trailing edges of $\overline{WE}$ can be independently stretched via CR6F_4-3 and MM8204_4-3.

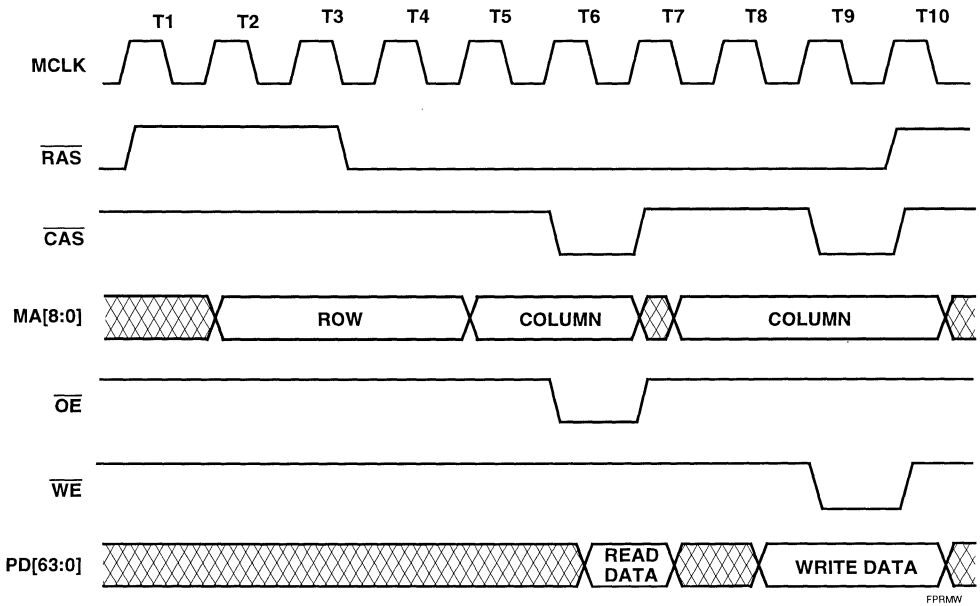


Figure 7-5. Fast Page Mode Read/Modify/Write Cycle

Figure 7-6 shows the functional timing for an Extended Data Out (EDO) mode read cycle. One difference between an EDO read cycle and a fast page mode read cycle is that EDO memory holds the data valid longer, allowing the data to be latched one cycle later (rising edges of T8 and T10). This allows the use of slower access time memory or a faster MCLK. Therefore, the last page access (or first for a single access) is one MCLK longer. Note that $\overline{RAS}$, the last $\overline{CAS}$ and $\overline{OE}$ are all stretched one MCLK and $\overline{OE}$ is held low for the entire cycle instead of being pulsed as in a fast page mode cycle.

The timing adjustments for $\overline{RAS}$, $\overline{CAS}$/$\overline{OE}$ and $\overline{WE}$ as described above for fast page mode cycles also apply to EDO cycles. Note that if the minimum $\overline{RAS}$ active time is specified as 3.5 MCLKs,

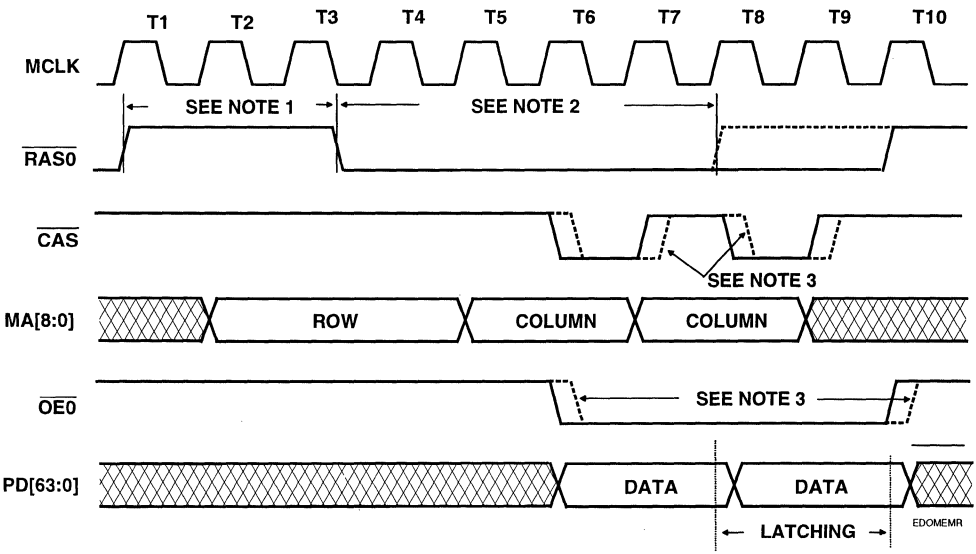


Figure 7-6. EDO Mode Read Cycle

Notes

1. The $\overline{RAS}$ precharge time can be adjusted via CR68_3, MM8204_1 and CR58_7.
2. The $\overline{RAS}$ low time for a single column access is adjustable via CR68_MM8204_2 and CR58_7. (The dashed line shows the $\overline{RAS}$ signal if the second page mode cycle were to be eliminated.)
3. The $\overline{CAS}$ and $\overline{OE}$ edges can be stretched via CR68_1-0 and MM8204_6-5. $\overline{CAS}$ and $\overline{OE}$ edges move together, but the leading and trailing edges can be stretched independently.

the actual minimum for a single EDO read cycle will be 4.5 MCLKs.

An EDO write cycle is functionally the same as a fast page mode write cycle.

Figure 7-7 shows the functional timing for an EDO mode read/modify/write cycle. Since read data is latched later than for a fast page mode cycle, there is less time available between the read and write.

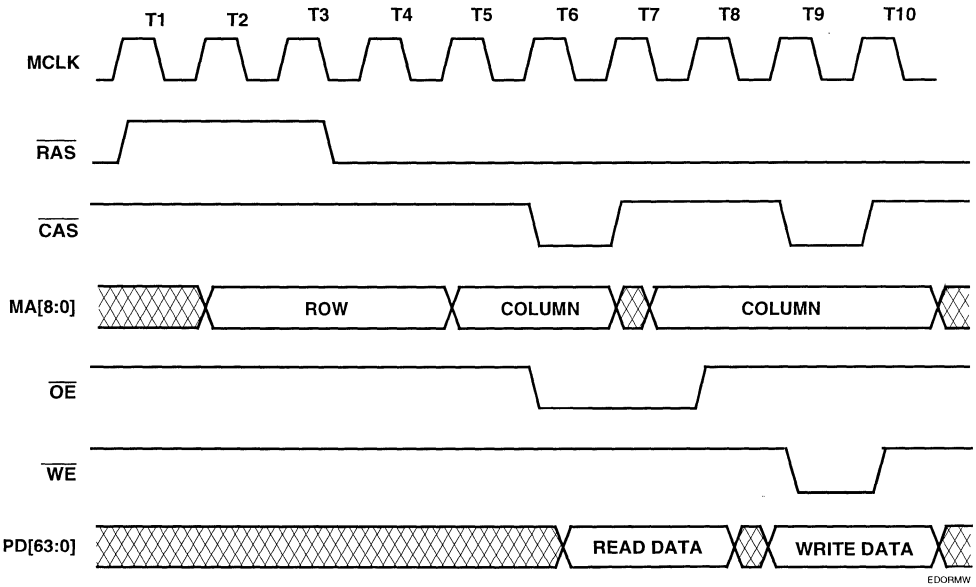


Figure 7-7. EDO Mode Read/Modify/Write Cycle

7.4 1-CYCLE EDO OPERATION

Bits 3-2 of CR36 are cleared to 00b to indicate that 1-cycle EDO DRAM operation is being used.

The functional timing for 1-cycle EDO reads is provided by Figure 7-8. The DRAM drives valid read data after the $\overline{CAS}$ falling edge at T5. The chip latches the data on the next falling $\overline{CAS}$ edge. Note that a dummy cycle is required at the end to latch the last read.

The functional timing for 1-cycle EDO writes is shown in Figure 7-9. Write data is latched by the DRAM on the falling edge of $\overline{CAS}$. No dummy cycle is required.

Figure 7-10 shows a read/modify/write cycle with 1-cycle EDO operation. A dummy cycle is added between the read and write.

CPU (i.e., linear addressing) access to memory and hardware cursor fetching are not supported with 1-cycle EDO. 2-cycle EDO operation will automatically be used for these functions.

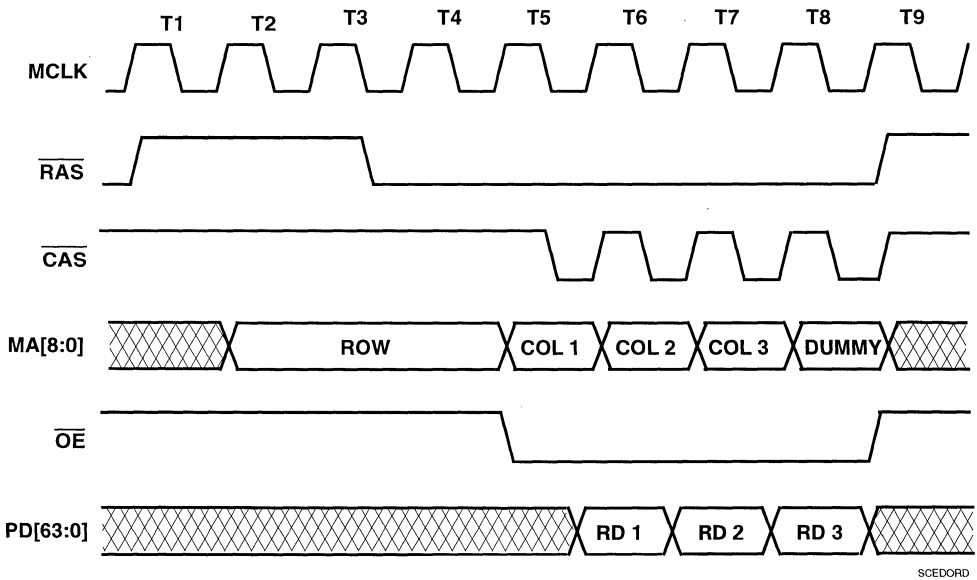


Figure 7-8. 1-Cycle EDO Mode Read

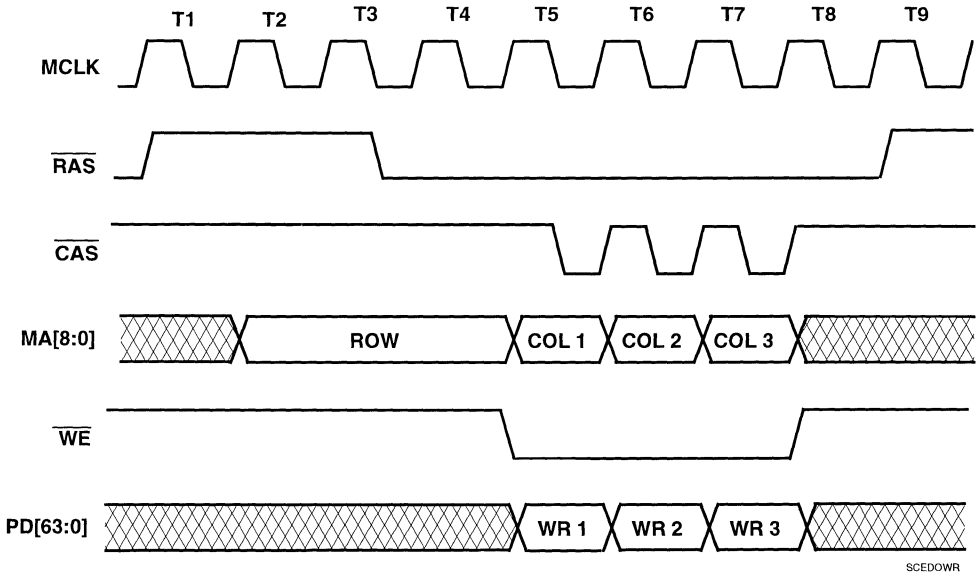


Figure 7-10. 1-Cycle EDO Mode Write

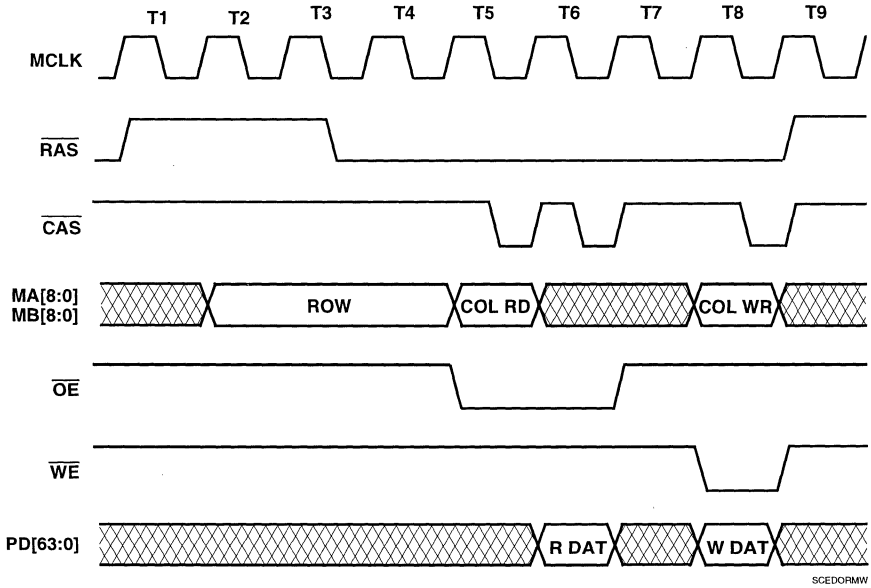


Figure 7-9. 1-Cycle EDO Mode Read/Modify/Write

7.5 VRAM TRANSFER OPERATION

Figure 7-11 shows the functional timing for a read transfer cycle. This cycle is defined by $\overline{CAS}$ and $\overline{WE}$ being high and $\overline{OE}$ and $\overline{DSF}$ being low at the falling edge of RAS. This operation loads the data from one row of the DRAM side of the VRAM into the SAM (serial access memory) register on the serial out side of the VRAM. The address latched on the falling edge of $\overline{RAS}$ specifies the DRAM row to be transferred. The SAM start address (the starting point for serial output) is latched on the falling edge of $\overline{CAS}$. The transfer is complete by the rising edge of $\overline{OE}$.

The amount of data transferred each cycle is a function of the VRAM design. Each VRAM type will specify a SAM size, either 256 or 512. This is the number of "words" per transfer, with the "word" size being equal to the width of the DRAM. For example, a x8 VRAM with a 512 SAM size transfers 512x8 bits per VRAM chip. With 64-bit operation, a total of 4096 bytes are transferred each cycle, assuming serial addressing. This is doubled for parallel addressing. Bit 6 of CR58 specifies the SAM size as either 512 (=0) or 256 (=1). If the designer is unsure of the SAM size to be used, this bit should be set for 256. A setting of 512 can enhance performance if the VRAM can support it.

ViRGE/VX always performs a full transfer cycle during each horizontal blanking period. CR3B plus bit 6 of CR5D define the horizontal character position where this transfer begins. This specification is enabled by setting bit 4 of CR34 to 1. Split transfers are enabled by default. They can be disabled by setting bit 6 of CR51 to 1, but this is not recommended for any mode.

Figure 7-12 shows the functional timing for a split transfer cycle. The cycle specification is the same as for a full transfer except that DSF is high on the falling edge of $\overline{RAS}$. During a split transfer, only 1/2 the data is transferred from the DRAM to the SAM as would be transferred for a full transfer.

For a split transfer, the SAM is divided into an A half and a B half. Immediately after the full transfer during blanking, the A half of the SAM is replaced (A1 in Figure 7-13). When enough data has been shifted out to reach the midpoint of the A half of the SAM (the small s point in the A1 block of Figure 7-13), the data in the B half (B1 in Figure 7-13) of the SAM is replaced concurrently with the shifting out of the rest of the data in the A half. Similarly, when half the data has been

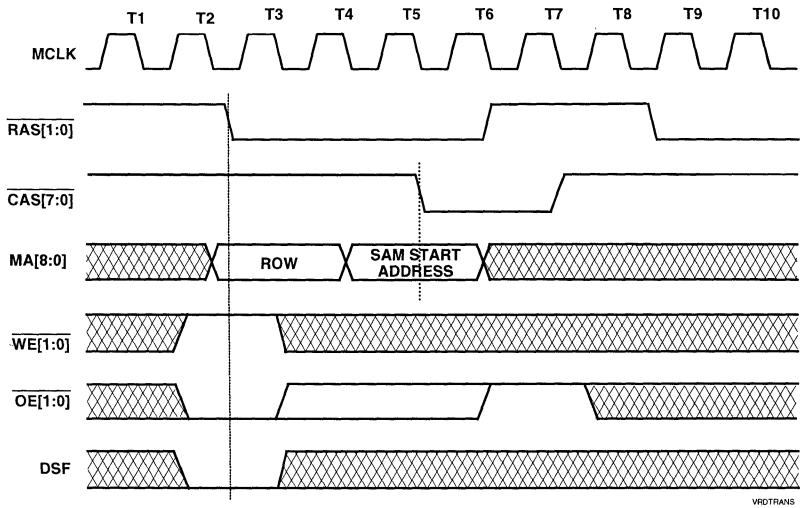


Figure 7-11. Full Read Transfer Cycle Timing

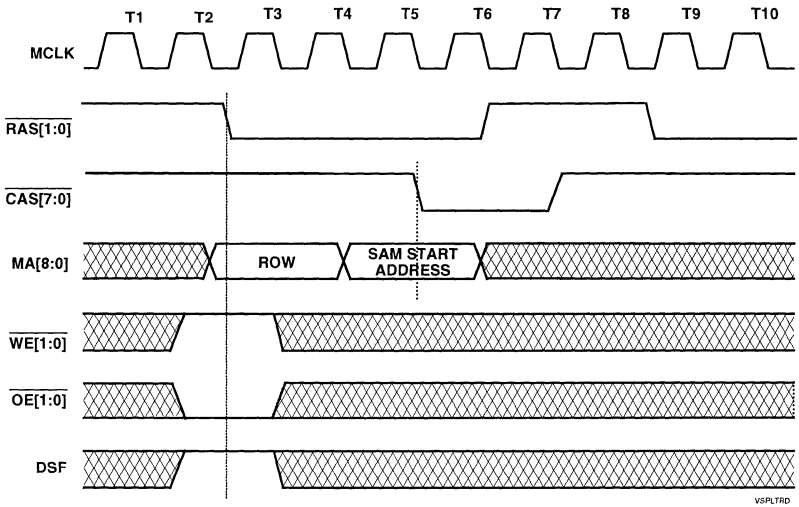


Figure 7-12. Split Read Transfer Cycle Timing

shifted out of the B half of the SAM (the small s in the B1 block of Figure 7-13), a split transfer to replace the data in the A half (A2) is initiated. B2 is then filled when the midpoint of A2 is reached.

In this example, the end of the active screen period for the first line is reached when some part of B2 has been shifted out. ViRGE/VX then performs another full transfer and another split transfer into the A half of the SAM during the horizontal blanking period. When blanking ends, screen refreshing for the next line is continued from the point (in B2) where it was stopped on the previous line. The rest of the line is then refreshed (A3, B3, part of A4) and the full transfer, split transfer sequence is repeated.

A full transfer during the active screen period will cause screen corruption. Split transfers allow even the longest screen lines (which contain more data than can be held in one SAM) to be refreshed without causing this problem.

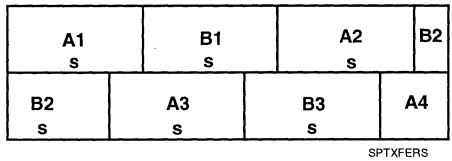


Figure 7-13. Split Transfers

7.6 VRAM SID OPERATION

Figure 7-14 shows the functional timing for the VRAM SID port when using a single 64-bit SID bus (CR66_5-4 = 00b or 01b).

$\overline{SE}$ is driven low from a rising clock edge. Data is clocked out by the next rising edge of SC after $\overline{SE}$ falls. Figure 7-14 shows the fourth data quadword coming from another bank. For example, if $\overline{SE0}$ and SC0 are active, 64 bits are driven from the first 2 MBytes each SC. If $\overline{SE0}$ is then deasserted and $\overline{SE2}$ asserted, the next data unit will come from the third 2 MBytes of memory. Note that $\overline{SE0}$ or $\overline{SE2}$ go with SC0 and $\overline{SE1}$ and $\overline{SE3}$ go with SC1.

Figure 7-14 and the above discussion assume 8 MBytes of memory. For a 4-MByte configuration, the $\overline{SE2}$ and $\overline{SE3}$ signals are not used.

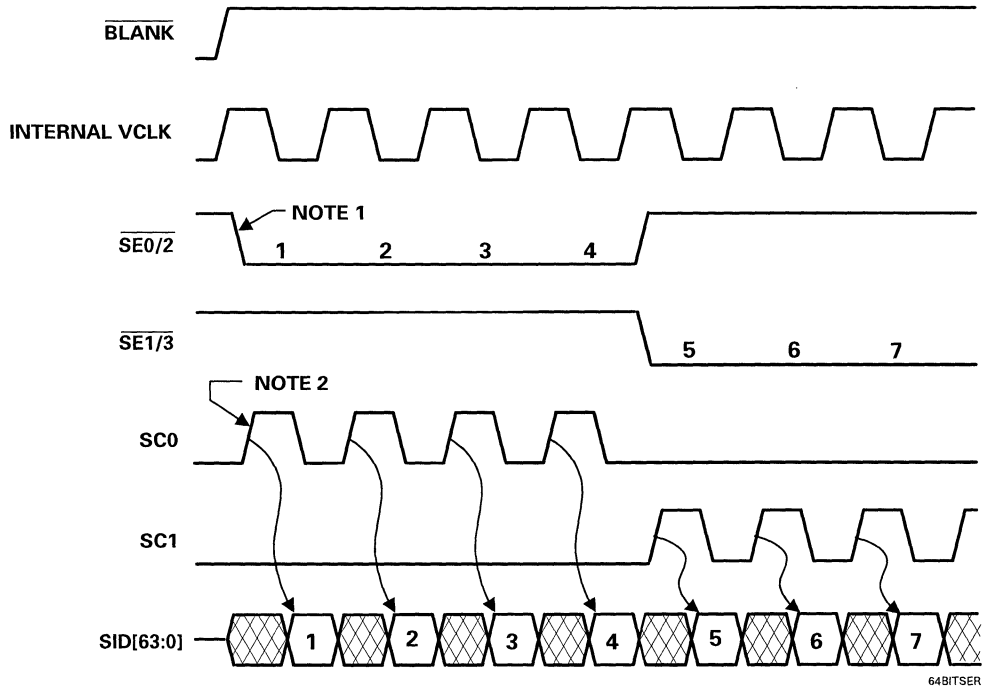


Figure 7-14. 64-bit Serial VRAM SID Timing

Notes

1. The falling edges of all the $\overline{SE}$ signals can be delayed (a few nanoseconds) via CR65_0. Care must be taken that this does not push this edge too close to SC rising, which must trail $\overline{SE}$ falling.
2. SC can be delayed in ICLK units via CR6D_6-4. $\overline{BLANK}$ can also be delayed via CR6D_2-0 so that the relationship between SC and $\overline{BLANK}$ can be controlled.

Figure 7-15 shows the functional timing for the SID port when using the 128-bit parallel mode. Two quadwords are driven out (to two separate SID busses - see Figure 7-2) each SC. The figure shows a bank switch for the fourth quadword. This mode is only used for 1600x1200x24 with an external 128-bit RAMDAC.

7.7 BLOCK WRITE SUPPORT

VRAMs supported by ViRGE/VX provide a block write feature. Each RAM contains a color register, which is programmed before a block write is performed. During the block write, each data bit written to the PD bus is expanded to one byte. A bit of logic 1 is expanded to a byte with the color register value. A bit of logic 0 does not change the corresponding memory byte. Each block write writes 64 bits of data, corresponding to 64 bytes of memory data respectively.

Block write support is enabled by setting CR57_7 to 1. When enabled, block writes are activated automatically for those graphics operation/mode combinations for which this feature provides a

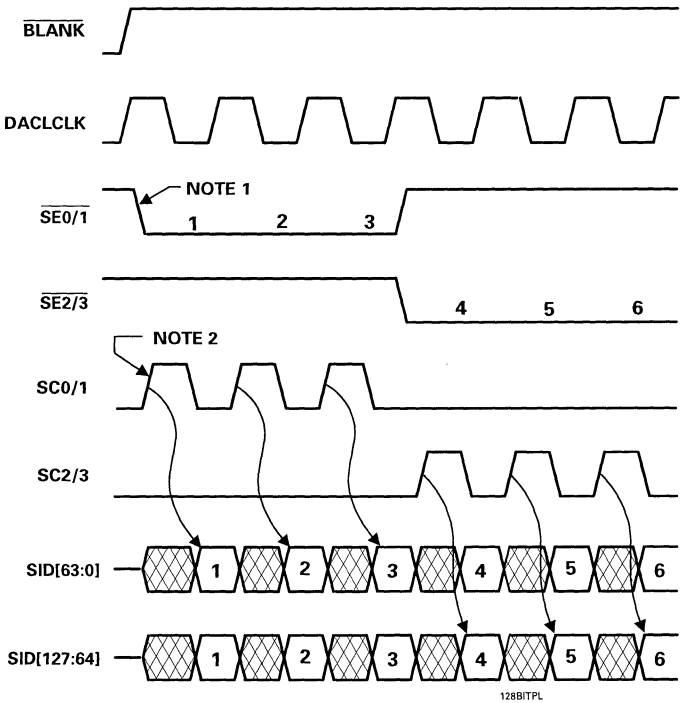


Figure 7-15. 128-bit SID VRAM Timing

Notes

1. The falling edges of all the $\overline{SE}$ signals can be delayed (a few nanoseconds) via CR65_0. Care must be taken that this does not push this edge too close to SC rising, which must trail $\overline{SE}$ falling.
2. SC can be delayed in ICLK units via CR6D_6-4. $\overline{BLANK}$ can also be delayed via CR6D_2-0 so that the relationship between SC and $\overline{BLANK}$ can be controlled.

performance benefit. These include solid color 2D rectangle and polygon fills and mono transparent BitBLTs in 8- and 16-bit modes. Color pattern BitBLTs are supported for 8-bit modes. Block writes are not supported for 24 bits/pixel modes.

When using block writes for BitBLTs, the transfer width is important. If the width is too small, block writes will cause a performance penalty instead of providing a benefit. CR57_5 = 0 specifies a minimum transfer width of 16 bytes for invoking block writes and CR57_5 = 1 specifies a minimum transfer width of 32 bytes.

Color pattern block writes require a transfer length of at least 64 bytes before they provide a performance benefit. If the transfer width for a given command will be less than 64 bytes, block write should be disabled for that command by setting bit 16 of the 2D Command Set Register (MMAx00) to 1. There is no minimum width checking for rectangle and polygon fills.

For all drawing commands using block writes, the drawing direction must be from left to right (bit 25 of the 2D Command Set register set to 1). In addition, the destination base address must be 64 bytes aligned (bits 5:0 are '0's). This must be detected by software, after which the block write enable bit can be set. If this is not done, garbage will be written to the screen. If the memory configuration contains DRAM, software must be careful to not attempt block writes to that DRAM.

7.8 DISPLAY MEMORY ACCESS CONTROL

A number of processes compete for access to display memory. These competing processes are (in decreasing order of access priority):

- Primary Stream High
- RAM refresh
- RAMDAC read/write
- Secondary Stream High
- Hardware cursor fetch
- LPB
- Read DMA High
- CPU accesses
- S3d Engine accesses
- Primary Stream Low
- Secondary Stream Low
- Read DMA Low

The three processes with high and low priorities have associated threshold register fields. If the current count is above the threshold level for a process, that process is given its low priority. Once the threshold is reached, the process is given its high priority.

Each of the processes (except for RAM refresh and hardware cursor fetch) has an associated timeout register field. These define the maximum latencies for giving up the memory bus when another process requests control. All of these threshold and timeout fields are described in the MPC Register section.

Section 8: RAMDAC Functionality

ViRGE/VX provides an internal 24-bit RAMDAC, with three 256 8-bit word color look-up table (LUT) RAMs feeding three 8-bit DACs. The maximum dot clock (DCLK) rate is 220 MHz for 8-, and 15/16-bit/pixel color modes (no Streams Processor operation) and 135 MHz for all 24-bits/pixel modes and all modes with the Streams Processor active. This support applies to all modes requiring 4 MBytes or less of video memory. The internal RAMDAC also provides gamma correction for 15/16- and 24-bits/pixel modes.

ViRGE/VX also supports a 128-bit external RAMDAC, allowing operation at a maximum DCLK rate of 220 MHz for all modes. This includes the 8-MByte mode (1600x1200x24). Streams Processor support is not available with an external RAMDAC, nor is gamma correction unless it is provided by the external RAMDAC.

8.1 OPERATING MODES USING THE INTERNAL RAMDAC

All the operating modes and the bit setting required to generate them are given in Table 8-1. This table also shows the basic relationships among the clocks controlling the flow of pixel data from memory to the RGB output. In Enhanced modes, SC clocks data from the VRAM SID port to ViRGE/VX. ICLK clocks the buffered input data to the internal RAMDAC. DCLK is the pixel rate. In VGA modes, the SID port is not used. Instead, pixel data is returned to ViRGE/VX via the PD lines.

8.1.1 VGA 8 Bits/Pixel

VGA operation is selected when CR63_0 = 0. For no Streams Processor operation, CR67_3-2 = 00b. All VGA modes are available with this setting. With Streams Processor operation, CR67_3-2 = 01b. Only VGA modes D, E, 10, 12 and 13 are supported.

In all cases, 8 pixel address bus bits are ANDed with the contents of the Pixel Read Mask register (3C6H) each DCLK. The result of the AND operation selects one of 256 LUT locations. This results in the output of 8 bits of color information to each of the three DACs or 24 bits to the Streams Processor.

8.1.2 8 Bits/Pixel Enhanced

Enhanced mode operation is selected when CR63_0 = 1. This enables use of the VRAM SID port and also the S3d Engine.

Table 8-1 shows that there are three 8 bits/pixel enhanced modes. The two 135 MHz modes are designed to be used together when switching Streams Processor operation on and off is desired. The DCLK remains unchanged, so the switch is transparent to the user. The 220 MHz mode provides higher performance, but if a switch is made to the 135 MHz Streams Processor mode, the DCLK must be re-programmed, resulting in screen flicker during the switch.

As with VGA modes, 8 pixel address bus bits are ANDed with the contents of the Pixel Read Mask register (3C6H) each DCLK. The result of the AND operation selects one of 256 CLUT locations. This

Table 8-1 Color Modes

Color Mode	MAX DCLK (MHz)	MAX ICLK (xDCLK)	SC (xDCLK)	CR63 Bit 0	CR67 Bits 3-2	CR67 Bits 7-4
VGA	Monitor Frequency	1	N/A	0	00	N/A
Enh. 8 bpp	135	1/2	1/8xA	1	00	0000
Enh. 8 bpp	220	1/4	1/8xA	1	00	0001
Enh. 15 bpp	135	1/2	1/4xA	1	00	0010
Enh 15 bpp	220	1/4	1/4xA	1	00	0011
Enh. 16 bpp	135	1/2	1/4xA	1	00	0100
Enh. 16 bpp	220	1/4	1/4xA	1	00	0101
Enh. 24 bpp	135	1/2	3/8xA	1	00	1101
VGA + SP	Monitor Frequency	1		0	01	N/A
Enh. 8 bpp + SP	135	1/2	1/8xA	1	11	0001
Enh. 15 bpp + SP	135	1/2	1/4xA	1	11	0011
Enh. 16 bpp + SP	135	1/2	1/4xA	1	11	0101
Enh. 24 bpp + SP	135	1/2	3/8xA	1	11	1101

SP = Streams Processor active
A = 1/2 when 2x scaling is used; otherwise it is 1

With an external RAMDAC, ICLK is the DACSCLK input and SC is the DACLCLK output.

results in the output of color information to the DACs or Streams Processor.

8.1.3 15/16-Bits/Pixel Enhanced

All the comments for 8 bits/pixel Enhanced operation are valid for these modes except that the CLUTs are not used. Instead, color data is passed directly to the DACs or Streams Processor.

8.1.4 24 Bits/Pixel Enhanced

Table 8-1 shows that 135 MHz is the maximum DCLK rate for true color operation. As with 15/16 bits/pixel, the CLUTs are not used and color data is passed directly to the DACs or Streams Processor.

8.2 GAMMA CORRECTION

Gamma correction mode is enabled by setting CR67_1 to 1. This can only be done for 15/16- and 24-bits/pixel modes and only for the primary stream if the Streams Processor is fully enabled. With Gamma correction enabled, the color data is taken from the three CLUTs, one byte per CLUT. CR55_6 must be set to 1 to provide 8-bit CLUT data. The CLUTs are programmed to provide color correction for monitor or printer inaccuracies, allowing correct color matching.

8.3 INTERNAL RAMDAC REGISTER ACCESS

The standard VGA RAMDAC register set (3C6H - 3C9H) is used to access the internal RAMDAC registers.

8.4 RAMDAC SNOOPING

Setting bit 5 of the PCI configuration space Command register (Index 04H) to 1 causes ViRGE/VX to snoop for RAMDAC writes. This means that ViRGE/VX will write the data to its local RAMDAC but will not claim the cycle by asserting DEVSEL. This allows the ISA controller to also generate a write cycle to a secondary RAMDAC. ViRGE/VX always provides the data for RAMDAC reads.

If bit 5 of the PCI Command register is cleared to 0, ViRGE/VX claims all RAMDAC read and write cycles.

Bits 2-0 of CR34 allow handling of PCI master aborts and retries to be individually enabled or disabled during RAMDAC cycles.

8.5 SENSE GENERATION

The internal RAMDAC contains analog voltage comparators. These drive the internal SENSE signal active low whenever the output on any of the AR, AG or AB pins exceeds 330 mV ± 20%. The state of this internal signal can be read via bit 4 of 3C2H. This information can be used to detect the existence and type of monitor (color/mono) connected to the system.

If an external RAMDAC is used, SENSE information is obtained via a register read.

8.6 POWER CONTROL

ViRGE/VX provides a PDOWN input pin. When a logic 0 signal is driven to this pin, the RGB analog outputs from the internal RAMDAC are turned off.

8.7 BLANK PEDESTAL

BLANK pedestal support is enabled by setting SR1A_5 to 1. This white level output remains at 17.6 mA (typical) above the black level output. However, the black level output is raised to 1.45 mA (typical) over the BLANK output level of 0 mA. RAMDAC output using the BLANK pedestal is recommended for some monitors.

8.8 EXTERNAL RAMDAC SUPPORT

Figure 8-1 shows the interface between ViRGE/VX and an external RAMDAC. Use of an external RAMDAC is enabled by setting CR55_3 to 1. Since the pixel data for enhanced modes goes directly to the RAMDAC (bypassing ViRGE/VX), no Streams Processor functions are available and gamma correction is not available unless supported by the external RAMDAC. In VGA modes, the pixel data is sent to the RAMDAC via the feature connector interface (PA[7:0]).

Because of the limitation described above, an external RAMDAC will normally be used as an option for the highest resolutions. Specifically, this is required for the one supported 8-MByte mode (1600x1200x24). It can also be used to provide higher refresh rates for other 24 bits/pixel modes, which are restricted to a maximum 135 MHz DCLK when using the ViRGE/VX internal RAMDAC. Because of the high performance requirements, a 128-bit 220 MHz SID RAMDAC is recommended.

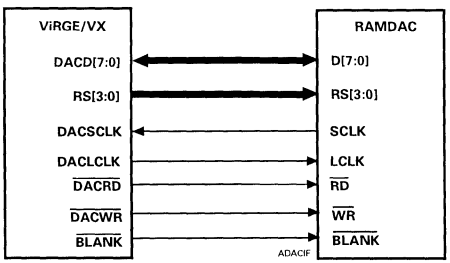


Figure 8-1. External RAMDAC Interface

8.9 RAMDAC SIGNATURE ANALYSIS

RAMDAC signature analysis is a method of testing that the video data pipeline is operating correctly.

8.9.1 Signature Analysis Steps

1. Set up the desired mode and eliminate all borders (BLANK active during the entire non-active display period). To do this, set CR33_5 to 1. This does not quite work for the right border, so CR6D is used to delay BLANK and SC[3:0] to produce the desired condition.
2. Wait until the VSYNC period. 3DAH_3 = 1 indicates active VSYNC. Alternately, set CR32_4 = 1 to enable interrupts. Set CR11_4 = 1 and CR11_5 = 0 to enable the VSYNC interrupt. Then poll for 3C2H_7 = 1. (The following two steps must take place during the blanking period. Using the VSYNC period is safe and this is easier to detect. The test could also be run between HSYNCs, but these are harder to detect.)
3. Reset the signature register by toggling SR18_1 from low to high to low.
4. Set SR18_0 to 1 to enable clocking of the signature register.
5. Generate a known and repeatable sequence of pixels to the RAMDAC during the next non-blanking period. Each pixel is logically mixed with the current signature register contents and the result is left in the signature register.
6. Wait for the next VSYNC active, then clear SR18_0 to 0 to disable clocking of the signature register.
7. Toggle SR18_2 from low to high to low to read out red signature data, then read this data from CR6E.
8. Toggle SR18_3 from low to high to low to read out green signature data, then read this data from CR6E.
9. Toggle SR18_4 from low to high to low to read out blue signature data, then read this data from CR6E.

The expected result for a given test can be manually calculated from the logic equations used to update the signature register. These are given below. In addition, all tests using the same input data stream should produce the same result.

8.9.2 Signature Generation

A byte of data (IN) is read in logically and mixed with a byte of all 0's. The result (Q) is stored and then mixed with the next IN byte, resulting in a new Q value (Qnext). This process is repeated until all data is read in, at which time the final Q value is inverted. This is the signature that is read from the signature register.

Bits 0-6 of each Q value are generated by the following logic:

Qnext[k-1] = XNOR(IN[k], Q[K]), k = 1 to 7

Bit 7 of each Q value is generated by the following logic:

Qnext[7] = XNOR(XNOR(IN[0], Q[0]), XNOR(IN[3], Q[3]))

Signature output = NOT Q (for the final Q byte)

Section 9: Clock Synthesis and Control

ViRGE/VX contains two phase-locked loop (PLL) frequency synthesizers. These generate the DCLK (video clock) and MCLK (memory clock signals for the graphics controller block.

9.1 CLOCK SYNTHESIS

Each PLL scales a single reference frequency input on the XIN pin. By placing a parallel-resonant crystal between the XOUT output pin and the XIN pin, the reference frequency is generated by ViRGE/VX's internal oscillator. Alternately, a CMOS-compatible clock input can be connected to XIN to provide the reference frequency.

The frequency synthesized by each PLL is determined by the following equation:

$$f_{OUT} = \frac{(M+2)}{(N+2)\times 2^R} \times f_{REF}$$

where:
R = 0, 1, 2 or 3 for MCLK
R = 0, 1, 2, 3 or 4 for DCLK

Programmed PLL M and PLL N values should be consistent with the following constraint:

$$220MHz \leq \frac{(M+2)f_{REF}}{(N+2)} \leq 440MHz \text{ (DCLK)}$$

Note that values used for the parameters are the integer equivalents of the programmed value. In particular, the R value is the code, not the actual frequency divisor.

The PLL M value can be programmed with any integer value from 0 to 127. The binary equivalent of this value is programmed in bits 6-0 of SR11 for the MCLK and in bits 6-0 of SR13, SR22 or SR24 for the DCLK. The PLL feedback loop frequency from the voltage controlled oscillator stage is scaled by dividing that frequency by (M+2).

The PLL N value can be programmed with any integer value from 0 to 31. The binary equivalent of this value is programmed in bits 4-0 of SR10 for the MCLK and in bits 4-0 of SR12, SR21 or SR23 for the DCLK. The reference frequency is divided by (N+2) before being fed to the phase detector stage of the PLL.

The PLL R value is a 3-bit range value that can be programmed with any integer value from 0 to 3 for MCLK or 0 to 4 for DCLK. The R value is programmed in bits 6-4 of SR 10 for MCLK and bits 7-5 of SR12, SR21 or SR23 for DCLK. This value codes the selection of a frequency divider for the PLL output. This is shown Table 8-1.

Table 9-1. PLL R Parameter Decoding

R-Range Code	Frequency Divider
000	1
001	2
010	4
011	8
100	16 (DCLK only)

The entire PLL block diagram is shown in Figure 8-1.

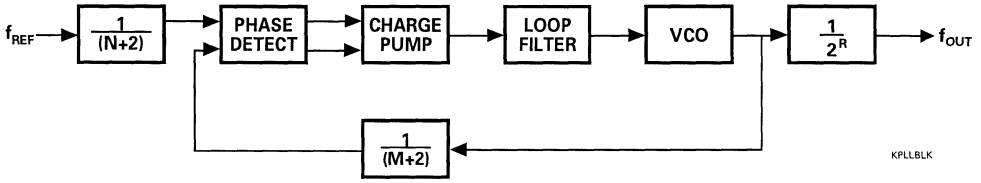


Figure 9-1. PLL Block Diagram

The following sequence may be followed to arrive at M and N values for any mode.

1. Calculate an R which does not violate the following constrains:

   $220MHz < 2^R \times f_{OUT} \leq 440MHz$

2. Start with N = 0 and calculate:

   $$M = [\frac{f_{OUT} \times (N+2) \times 2^R}{f_{REF}}] - 2$$

3. Determine if the following constraint is met:

   $$0.995\ f_{OUT} < \frac{(M+2)\ f_{REF}}{(N+2)\ 2^R} < 1.005\ f_{OUT}$$

4. If the constraint in step 3 is met, the M and N values used will generate the desired frequency (within the specified tolerance). If the constrain is not met, repeat steps 2 and 3 with N increased by 1 each time until the constraint in step 3 is met. Note that multiple combinations of M and N are possible for a given output frequency, but the best combination will be the one with the lowest N.

9.2 CLOCK REPROGRAMMING

If 3C2_3-2 = 00b, the DCLK PLL parameters are taken from SR22 and SR23. The default values generate a frequency of 25.175 MHz. This is the DCLK frequency generated at power on to support standard VGA operation.

If 3C2_3-2 = 01b, the DCLK PLL parameters are taken from SR24 and SR25. The default values generate a frequency of 28.322.

For Enhanced mode operation, 3C2_3-2 are programmed to 11b and the DCLK PLL values are taken from SR12 and SR13. No default values are defined for these registers.

New DCLK PLL parameter values can be programmed at any time. They can be loaded in one of two ways. If bit 5 of SR15 is cleared to 0, the new DCLK frequency is loaded by setting bit 1 of SR15 to 1. Bit 1 of SR15 should be left at a value of 1. Actual loading will be delayed for a short but variable period of time.

The alternate approach to loading the new DCLK frequency is to toggle bit 5 of SR15 by programming it to a 1 and then a 0. This immediately loads the DCLK (and MCLK) frequencies (no variable delay). For example, pseudocode to change DCLK to the frequency specified by PLL parameter values of 34H and 56H is:

```
3C2 ⇐ 6FH  ; DCLK specified by
           ; SR12 and SR13
3C4 ⇐ 12H  ; SR12 index
3C5 ⇐ 34H  ; SR12 PLL value
3C4 ⇐ 13H  ; SR13 index
3C5 ⇐ 56H  ; SR13 PLL value
3C4 ⇐ 15H  ; SR15 index
3C5 ⇐ RMW  ; Use read/modify/write to
           ; set bit 5 to 1 and leave
           ; other bits unchanged
3C5 ⇐ RMW  ; Use read/modify/write to
           ; clear bit 5 to 0 and
           ; leave other bits
           ; unchanged
```

Either loading approach should work. The second (immediate loading) approach helps with system testing since the timing of the load is predictable. The first approach (via bit 1 of SR15)

has the advantage of separating the loading of DCLK from that of MCLK.

After power-up, all MCLK frequency changes must be made by re-programming SR10 and SR11. If bit 5 of SR15 is cleared to 0, the new frequency does not take effect until a 1 has been written to bit 0 of SR15. This bit must then be cleared to 0 to prevent repeated loading. Actual loading will be delayed for a short but variable period of time.

As explained above for DCLK, toggling bit 5 of SR15 (0,1,0) immediately loads both the DCLK values in the register pair selected by 3C2_3-2 and the MCLK values in SR10 and SR11.

9.3 CLOCK SELECTION AND CONTROL

DCLK is generated by the internal clock synthesizer. ICLK is the signal used to clock pixel data into the internal RAMDAC. For most modes of operation, ICLK is generated directly from DCLK and has the same phase (neglecting internal gate delays). Bit 0 of CR67 provides the option to invert DCLK before it becomes ICLK. CR66_2-0 provides the ability to make ICLK one of a number of fractions of DCLK.

The internal RAMDAC can also have pixel data clocked in by an externally provided feature connector clock. For the VESA Advanced Feature Connector (VAFC), this clock is VCLKI, which is selected via bit 1 of SRB. For a pass-through connector, this clock is input on the VCLK pin and is enabled by asserting the $\overline{\text{EVCLK}}$ signal and by clearing bit 3 of CR33 to 0.

Certain 4 bits/pixel modes require that DCLK be halved. This is the case for bit 6 of AR10 set to 1 and bit 4 of CR3A cleared to 0 and is enabled by setting bit 4 or SR15 to 1 and clearing bit 3 of CR33 to 0.

9.4 CLOCK TESTING

The procedure for testing clock synthesis is:

1. Program SR10/SR11 (MCLK) or SR12/SR13 (DCLK) to generate the desired frequency.
2. Select either MCLK or DCLK for testing via SR14_3. If DCLK is selected, also ensure that 3C2_3-2 = 11b to select SR12/SR13 as the source of the DCLK PLL parameters.
3. Clear the clock synthesizer counter by toggling SR14_4.
4. Set SR14_2 = 1 for some exact amount of time (Δt).
5. Read SR16 (most significant byte) and SR17 (least significant byte). This is the binary count of the number of clock cycles actually executed.
6. Calculate the expected result (Δt/clock period) and compare it with the value read in the previous step. The two should agree within 2 or 3 counts.

Section 10: Streams Processor

The S3 Streams Processor processes data from the graphics frame buffer, composes it and outputs the result to the internal DACs for generation of the analog RGB outputs to the monitor. The general data flow is shown in Figure 10-1. Note that the DAC shown in this figure is inside ViRGE/VX.

10.1 INPUT STREAMS

The processor can compose data from up to 3 independent streams as shown in Figure 10-1:

1. Primary Stream - RGB graphics data
2. Secondary Stream - RGB or YUV/YCbCr (video) data from another region within the frame buffer
3. Hardware Cursor - 64x64x2 cursor, either Microsoft or X-11 definition

Regardless of the input formats, the Streams Processor creates a composite RGB-24 (8.8.8) output to the DACs. This means that, for example, RGB-8 pseudo-color graphics data can be overlaid with true-color-equivalent (24 bits/pixel) video data. The result is improved video quality and/or reduced memory bandwidth requirements as compared with systems that require both graphics and video to be stored in the same frame buffer format. In certain modes, the

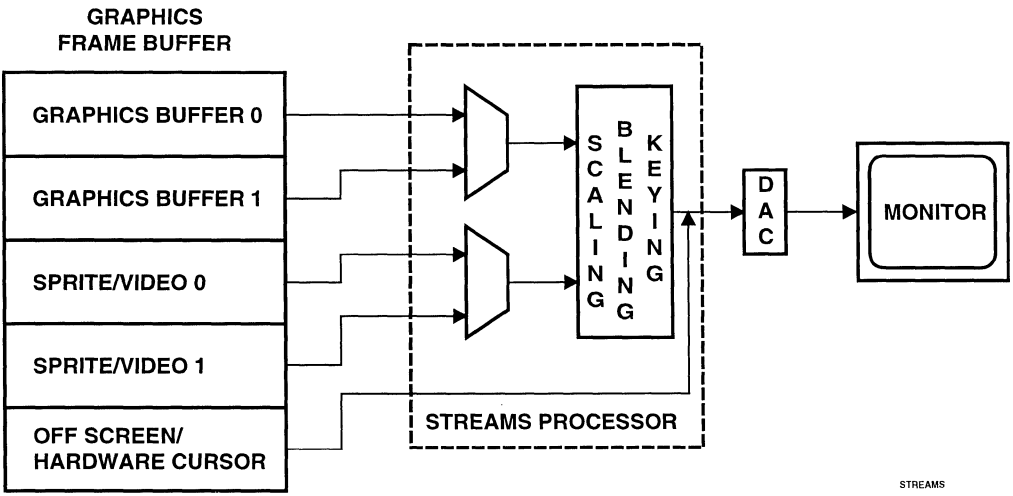


Figure 10-1. Streams Processor

Streams Processor also saves memory bandwidth by eliminating the need to save and restore the overlay background since the background (primary stream) is never overwritten in the frame buffer.

Streams Processor support is not available for interlaced graphics modes and standard VGA modes except for modes D, E, 10, 12 and 13.

Bits 3-2 of CR67 specify the Streams Processor mode of operation. If they are cleared to 00b, Streams Processor operation is disabled. They are programmed to 01b when the primary stream is VGA mode D, E, 10, 12 or 13 (the only supported modes). A secondary stream can be overlaid on the primary stream. CR67_3-2 are set to 11b to support an Enhanced mode primary stream and a secondary stream.

10.1.1 Primary Stream Input

The primary stream is generated by reading the RGB pixel data written to the frame buffer by the graphics controller. The format for this data can be any of the following as selected via bits 26-24 of MM8180.

- RGB-8 (Although not shown in Figure 1, the frame buffer data is first passed through the internal RAMDACs color lookup table (CLUT), where it is palettized before being passed to the Streams Processor.
- RGB-16 (5.6.5)
- RGB-24

10.1.2 Secondary Stream Input

The secondary stream is generated by reading pixel data from a separate section of the frame buffer than that used to generate the primary screen. This might be RGB data written by the graphics controller, such as a sprite used by game programmers for moving objects. It could also be RGB, YUV or YCbCr data written to the frame buffer by some video source (CPU, decoder, digitizer). The format for this data can be any of the following as selected via bits 26-24 of MM8190.

- YCbCr-16 (4.2.2), 16 - 240 input range
- YUV-16 (4.2.2), 0 -255 input range
- YUV (2.1.1)
- RGB-16 (5.6.5)
- RGB-24
- XRGB-32 (X.8.8.8) - X is the ignored upper byte.

The data can be passed through unscaled or scaled up horizontally and vertically by an arbitrary amount. YCbCr/YUV data is color space converted and all data is converted to RGB-24 (8.8.8) format.

10.1.3 Hardware Cursor Generation

Hardware cursor generation is explained in Section 15. The cursor is overlaid on the Streams Processor image.

10.1.4 Frame Buffer Organization/ Double Buffering

For each stream to be used, the starting location (offset) in the frame buffer and the stride (byte offset between vertically adjacent pixels on the screen) must be specified. Both the primary and the secondary streams can be double buffered as depicted in Figure 10-1. This means that duplicate frame buffer storage can be provided for both the primary and secondary image (or for either one of them). With double buffering, the programmer can rapidly switch from one primary or secondary image to the other. In addition, having two images allows more time for updating one image while the other is being displayed. Defining the frame buffer organization and implementing double buffering are done via the register fields described in Table 10-1. LPB stands for Local Peripheral Bus.

The secondary stream can be generated from data written to the frame buffer via the LPB when LPB mode is enabled. In this case, the Secondary Display Buffer Address 0 and the LPB Frame Buffer Address 0 will normally be the same, as will the Address 1's for both the secondary stream and the LPB input if double buffering is used. The various LPB control bits described in

Table 10-1. Register Fields Used For Specifying Frame Buffer Organization and Double Buffering

Register Field	Description
MM81C0_21-0	Primary Display Buffer Address 0. This is the starting address (offset) in the frame buffer for 1 primary graphics image.
MM81C4_21-0	Primary Display Buffer Address 1. This is the starting address (offset) in the frame buffer for a second primary graphics image.
MM81C8_11-0	Primary Stream Stride. This is the byte offset in the frame buffer from a pixel in a given primary image display line to the pixel directly below it on the next display line. The stride must be the same for both primary buffers.
MM81CC_0	Primary Stream Buffer Select 0 = Primary frame buffer starting address 0 (MM81C0_21-0) used for primary stream 1 = Primary frame buffer starting address 1 (MM81C4_21-0) used for primary stream
MM81CC_2-1	Secondary Stream Buffer Select 00 = Secondary frame buffer starting address 0 (MM81D0_21-0) used for the secondary stream 01 = Secondary frame buffer starting address 1 (MM81D4_21-0) used for the secondary stream 10 = Secondary frame buffer starting address 0 (MM81D0_21-0) used for the secondary stream and LPB frame buffer starting address 0 (MMFF0C_21-0) used for the LPB input stream OR secondary frame buffer starting address 1 (MM81D4_21-0) used for the secondary stream and LPB frame buffer starting address 1 (MMFF10_21-0) used for the LPB input stream. Which alternative applies is determined by LPB starting address register selected by bit 4 of this register. 11 = Secondary frame buffer starting address 0 (MM81D0_21-0) used for the secondary stream and LPB frame buffer starting address 1 (MMFF0C_21-0) used for the LPB input stream OR secondary frame buffer starting address 1 (MM81D4_21-0) used for the secondary stream and LPB frame buffer starting address 0 (MMFF10_21-0) used for the LPB input stream. Which alternative applies is determined by the LPB starting address register selected by bit 4 of this register.
MM81CC_4	LPB Input Buffer Select 0 = LPB frame buffer starting address 0 (MMFF0C_21-0) used for LPB input 1 = LPB frame buffer starting address 1 (MMFF10_21-0) used for LPB input
MM81CC_5	LPB Input Buffer Select Loading 0 = The value programmed in bit 4 of this register takes effect immediately 1 = The value programmed in bit 4 of this register takes effect at the end of the next frame (completion of writing all the data for a frame into the frame buffer)
MM81CC_6	LPB Input Buffer Select Toggle 0 = End of frame (completion of writing all the data fro a from into the frame buffer) has no effect on the setting of bit 4 of this register 1 = End of frame causes the setting of bit 4 of this register to toggle

Table 10-1. Register Fields Used For Specifying Frame Buffer Organization and Double Buffering (continued)

Register Field	Description
MM81D0_21-0	Secondary Display Buffer Address 0. This is the starting address (offset) in the frame buffer for 1 secondary graphics or video image.
MM81D4_21-0	Secondary Display Buffer Address 1. This is the starting address (offset) in the frame buffer for a second secondary graphics or video image.
MM81D8_11-0	Secondary Stream Stride. This is the byte offset in the frame buffer from a pixel in a given secondary image display line to the pixel directly below it on the next display line. The stride must be the same for both secondary buffers.
MMFF0C_21-0	LPB Frame Buffer Address 0. This is the starting address (offset) in the frame buffer for one image buffer into which is written data from the LPB. The secondary stream can be generated from this buffer.
MMFF10_21-0	LPB Frame Buffer Address 1. This is the starting address (offset) in the frame buffer for a second image buffer into which is written data from the LPB. The secondary stream can be generated from this buffer.
MM81CC_6	LPB Input Buffer Select Toggle 0 = End of frame (completion of writing all the data fro a from into the frame buffer) has no effect on the setting of bit 4 of this register 1 = End of frame causes the setting of bit 4 of this register to toggle

Table 10-1 allow complete hardware control of the capture and display of video data using either single or double buffering.

10.2 INPUT PROCESSING

Different processing options are available for the primary and secondary input streams. These are explained next.

10.2.1 Primary Stream Processing

The primary stream input RGB format is converted (if required) to RGB-24 (8.8.8) format. Each color byte is padded as required with low order zeros. After this conversion, the data can be passed through unscaled or scaled up horizontally and vertically by a factor of 2 via bits 30-28 of MM8180. For MM8180_30-28 = 001, horizontal scaling is done via replication. If these bits are programmed to 010, horizontal scaling is done using interpolation. If MM81E8_15 = 0, vertical scaling is done via replication. If MM81E8_15 = 1, vertical scaling is done via interpolation. The 2x scaling allows a 320x240 image (as used by many games) to be displayed at a full-screen 640x480 resolution.

10.2.2 Secondary Stream Processing

The secondary stream input format is converted (if required) to RGB-24 (8.8.8) format. For YUV/YCbCr inputs, the required color space conversion is automatically performed. Before conversion, the data can be passed through unscaled or scaled up horizontally and/or vertically by arbitrary factors. Horizontal scaling uses filtering for interpolation. Vertical scaling uses line replication for filtering (interpolation). Interpolation requires that twice the amount of data be processed, so there is a performance penalty that may limit scaling ratios. The register fields involved in scaling up the secondary stream are described in Table 10-2. Figure 10-2 graphically describes the various fields.

For example, assume a 10x10 window that is to be scaled up horizontally by a factor of 2.5. The filter characteristics are set for bi-linear (2x to 4x stretch). The starting line width is 10 pixels and the ending line width is 25 pixels. The DDA horizontal accumulator initial value is 2 (10-1) - (25-1) = -6. The K1 horizontal factor is 10-1 = 9. The K2 horizontal factor is 10-25 = -15. Programming these parameters with these values results in a 2.5x horizontal stretch for the secondary stream window.

10.3 COMPOSITION/OUTPUT

A variety of output types can be composed from the streams described above. The compose modes are:

1. MM81A0_26-24 = 000b - Secondary stream overlaid on the primary stream in an opaque rectangular window. This is the default mode and can be used, for example, for a video window overlaying the graphics screen. Note that this mode will not work for the case where the user needs to pull down a graphics window over the video since the graphics window is defined as being under the video window. Color keying (number 5 in this list) must be used for this purpose.
2. MM81A0_26-24 = 001b - Primary stream overlaid on the secondary stream in an opaque rectangular window. This could be used, for example, to provide graphics captions for a video window. The video is not visible behind the rectangular graphics window.
3. MM81A0_26-24 = 010b - Secondary stream blended with the primary stream on a pixel by pixel basis within the secondary stream window. This is used to provide a dissolve between two scenes.
4. MM81A0_26-24 = 011b - Secondary stream blended with the primary stream on a pixel by pixel basis within the secondary stream window. This is used to provide a fade between two scenes.
5. MM81A0_26-24 = 101b - Secondary stream overlaid on the primary stream in an irregular window. This requires a color key. This would be used, for example, for game sprites. Only the graphics area behind the sprite shape would be covered up.
6. MM81A0_26-24 = 110b - Primary stream overlaid on the secondary stream in an irregular window. This requires a color/chroma key. This case allows, for ex-

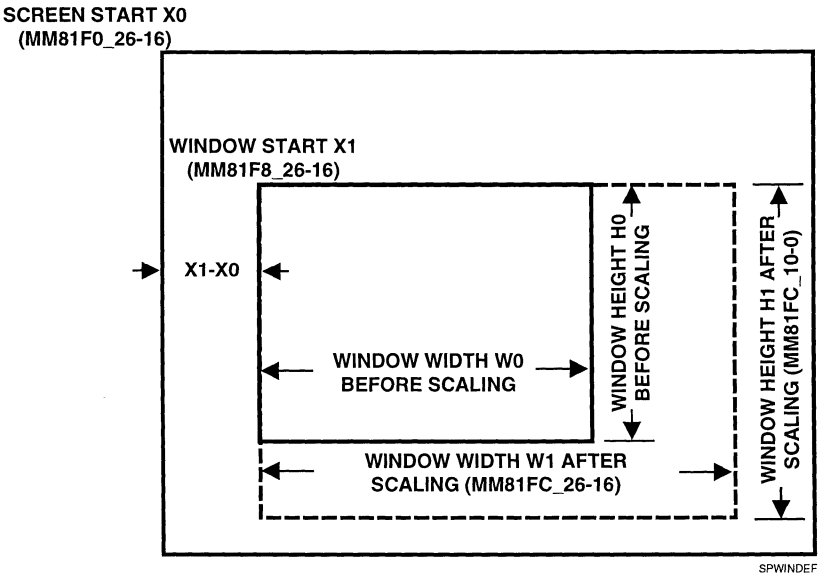


Figure 10-2. Screen Definition Parameters

Table 10-2. Register Fields Used For Scaling Up the Secondary Stream

Register Field	Description
MM8190_30-28	Filter Characteristics 000 = Secondary stream (pass-through) 001 = Secondary stream, linear, 0-2-4-2-0, for X stretch 010 = Secondary stream, bi-linear, for 2X to 4X stretch 011 = Secondary stream, linear, 1-2-2-2-1, for 4X stretch This selection applies only to horizontal scaling.
MM8190_11-0	DDA Horizontal Accumulator Initial Value. Value = 2 (W0-1) - (W1-1), where W0 is the line width in pixels before scaling and W1 is the line width after scaling. This is a signed value.
MM8198_10-0,	K1 Horizontal Factor. Value = W0-1, where W0 is the line width in pixels before scaling. This is a signed value.
MM8198_26-16	K2 Horizontal Factor. Value = W0-W1, where W0 is the line width in pixels before scaling and W1 is the line width after scaling. This is a signed value.
MM81E0_10-0,	K1 Vertical Factor. Value = [height (in lines) of the initial output window (before scaling)] -1. The initial output window height is the vertical resolution of the data written to the frame buffer and is shown as H0 in Figure 10-2.
MM81E4_10-0,	K2 Vertical Factor. Value = 2's complement of [height (in lines) of the initial output window (before scaling)] - [height (in lines) of the final output window (after scaling)] The initial output window height is the vertical resolution of the data written to the frame buffer and is shown as H0 in Figure 10-2. The final value is the same height value that is programmed in MM81FC_10-0 and is shown as H1 in Figure 10-2. This value is then the 2's complement of (H1 - H0).
MM81E8_11-0,	DDA Vertical Accumulator Initial Value. Value = 2's complement of [height (in lines) of the output window after scaling] -1. This is the same height value that is programmed in MM81FC_10-0 and is shown as H1 in Figure 10-2.
MM81E8_15	Vertical Scaling Type. 0 = line replication; 1 = interpolation

ample, graphics text to overlay video with the video appearing around and even inside of the text characters.

10.3.1 Opaque Rectangular Overlaying

These modes are items 1 and 2 in the compose modes list. When one of these modes is used, the programmer can invoke a feature called opaque overlay control. This is enabled by setting MM81DC_31 to 1. If MM81A0_26-24 = 000b (secondary stream on top), then MM81DC_30 must be cleared to 0 to also specify secondary stream on top. Similarly, If MM81A0_26-24 = 001b (primary stream on top), then MM81DC_30 must be set to 1 to also specify primary stream on top. The next step is to define when to stop fetching pixels for a line from memory and when to restart fetching them. The goal is to not fetch those pixels in the background window that are covered up by the opaque rectangular overlay window, thus saving memory bandwidth.

The first pixel that does not need to be fetched is at horizontal position X1 shown in Figure 10-2. This is programmed in MM8158_26-16. The starting pixel position for the background (X0) is programmed in MM81F0_26-16. The difference (X1 - X0) must be converted into quadwords and then programmed in MM81DC_12-3. The value is (X1 - X0) x bytes per pixel/8. If the result is a fraction, it is rounded up to the next highest integer to ensure that the first pixel not fetched in inside the opaque overlay window. Note that

if the secondary stream is in the background, then the value is (X0 - X1) x bytes per pixel/8, again rounded up.

Pixel fetching must start again before or at the last pixel position of the opaque overlay window. Using the terms in Figure 10-2, this position is (X1 - X0) + W1, with W1 programmed in MM81FC_26-16 (secondary stream is on top). Converting to quadwords, the value is [(X1 - X0) + W1] x bytes per pixel/8. If the result is a fraction, the result is truncated to the next lowest integer (minus 1) and programmed in MM81DC_28-19. Note that if the secondary stream is in the background, then (X0 - X1) is used and W1 is the value in MM81F4_26-16 (primary stream is on top).

Opaque overlay control cannot be used with keying or blending and should never be enabled when one of these modes is being used.

10.3.2 Blending

These modes are items 3 and 4 in the compose modes list. The blender accepts the primary and secondary pixel streams and blends them with an arithmetic weighting. The result is then overlaid with the cursor stream. Both blender inputs are RGB 8.8.8 from the outputs of the primary stream interpolator and secondary stream color space converter. Note that blending makes sense only when both streams are defined. In addition, when blending is selected, the concept of background/foreground or top and bottom window has no meaning.

Two types of blending are provided: dissolve and fade.

When dissolve is chosen, the output pixels are generated using the following equation:

[Pp x Kp + Ps x (8 - Kp)]/8

Pp and Ps are the primary and secondary stream pixel colors respectively, both RGB 8.8.8. Kp is the primary stream weighting factor. It is a 3-bit value programmed in MM81A0_12-10. This weight value is applied to each of the three color values for the pixel. If Kp = 0, only the secondary stream is displayed. As Kp is increased, more of the pixel color from the primary stream is blended into output. At the maximum (Kp = 7 or 111b), 7/8ths of the color will be due to the primary stream and 1/8th will be due to the secondary stream. Therefore, by starting with the primary stream only, then overlaying the secondary stream with Kp values decreasing from 7 to 0, the overlay window can be dissolved gradually from primary stream to secondary stream. Note that when the Kp value is reprogrammed, its new value does not take effect until the next VSYNC, so it can be reprogrammed during frame display without disruptive effects.

When fade is chosen, the output pixels are generated using the following equation:

[Pp x Kp + Ps x Ks]/8, where Kp + Ks must be ≤ 8.

Ks is the secondary stream weighting factor. It is a 3-bit value programmed in MM81A0_4-2. This weight value is applied to each of the three color values for the pixel. Note that when fading is selected, the default values for Kp and Ks (both 0) result in a color value of 0. As with Kp, when the Ks value is reprogrammed, its new value does not take effect until the next VSYNC.

10.3.3 Color/Chroma Keying

These modes are items 5 and 6 in the compose modes list. Keying is a way of selecting on a pixel by pixel basis which stream will be displayed. Color keying is used when the stream source is in RGB format (graphics). This is always the case for the primary stream. Chroma keying is used when the stream source is YUV or YCbCr (video). The secondary stream source can be either graphics or video, so either color or chroma keying might be used. If 81A0_26-24 (compose mode) = 101b and MM8184_28 = 1, the color key is compared with the primary stream pixel. If there is a match, the corresponding secondary stream pixel is displayed. If 81A0_26-24 = 110b and MM8184_28 = 1, the color or chroma key is compared with the secondary stream pixel. If there is a match, the corresponding primary stream pixel is displayed.

For RGB input types (as specified by MM8180_26-24), a color key must be defined. This is done by programming MM8184_23-0 with a specific RGB 8.8.8 color value. MM8184_28 must be set to 1 to

enable use of this value. The number of bits to compare for each color is specified in MM8184_26-24. If there is a color match with the keyed stream pixel, the corresponding other stream pixel is displayed.

If the secondary stream input format is YUV or YCbCr, the chroma key is specified as a range of color values. The lower bound value is defined in MM8184_23-0. The upper bound value is defined in MM8194_23-0. If the secondary stream pixel color value falls within this range (inclusive of the lower and upper bounds), the Streams Processor displays the corresponding pixel from the primary stream. If the secondary stream pixel color is outside this range, the secondary stream pixel is displayed.

10.3.4 Window Location

The starting X,Y coordinates and window size for the primary stream are specified in MM81F0 and MM81F4 respectively. The starting X,Y coordinates and window size for the secondary stream are specified in MM81F8 and MM81FC respectively.

10.4 STREAMS FIFO CONTROL

The streams FIFO can be reconfigured to optimize performance for various operating modes. The FIFO is 24 8-byte slots deep. For VGA plus secondary stream mode (CR67_3-2 = 01b), the FIFO can be reconfigured via MM8200_4-0 to assign all 24 slots to either the primary or secondary stream. Allocations of 16-8 and 12-12 slots between the two streams are also possible.

In full streams mode (CR67_3-2 = 11b) and with vertical filtering disabled, MM8200_4-0 should be programmed to allot 24 slots to the secondary stream. In full streams mode and with vertical filtering enabled (MM81E8_15 = 1), MM8200_4-0 should be programmed to allot 12 slots to the secondary stream and 12 to the primary stream.

No matter what the allocation, FIFO thresholds must be specified for the primary and secondary streams. This is done via MM8200_16-12 for the primary stream and MM8200_10-6 for the secondary stream. When the FIFO empties to the threshold level, an internal signal is generated requesting the memory controller to begin refilling the FIFO. The programmed threshold levels must never exceed the corresponding FIFO depths. The optimal settings for the threshold levels will be system and operating mode dependent and will have to be determined by trial and error.

10.5 VERTICAL FILTERING

ViRGE/VX does vertical (or Y direction) filtering when MM81E8_15 = 1. For each display line, two adjacent frame buffer lines are fetched into the streams buffer(s), then passed to a two-tap filter for bilinear filtering. This provides a higher quality image with some performance penalty. If MM81E8_15 = 0, line replication is used.

Vertical filtering is only available in the full streams processor operation mode (CR67_3-2 = 11b). It is not supported in the VGA plus secondary stream mode (CR67_3-2 = 01b).

Section 11: Local Peripheral Bus

The Local Peripheral Bus (LPB) function provides the following:

- S3 Scenic Highway interface to the Scenic/MX2 MPEG Audio/Video Decoder (glueless, bi-directional)
- Scenic Highway interface to the Philips® video digitizers (glue logic is required to convert 16-bit output to 8-bit ViRGE/VX input for VL-Bus configurations. However, the Scenic/MX2 has a glueless SAA7100 interface which can be used to provide the 16- to 8-bit conversion). A 16-bit data interface is available on ViRGE/VX for PCI configurations.
- Host Video Data Pass-through. This allows decimation of 32-bit CPU data being written to the frame buffer.)
- LPB Feature Connector (glueless 8-bit bi-directional or 16-bit VAFC)
- 4-bit General Input Port and 4-bit General Output Port

The Scenic Highway interface is clocked by LCLK. This requires that MMFF00_24 be set to 1. Pass-through operation is clocked by SCLK by clearing MMFF00_24 to 0.

The LPB mode also provides the support required for DDC2 monitor communications. This, the feature connector interfaces and the General Input/Output Port are described in Section 12.

The internal block diagram for the LPB is shown in Figure 11-1.

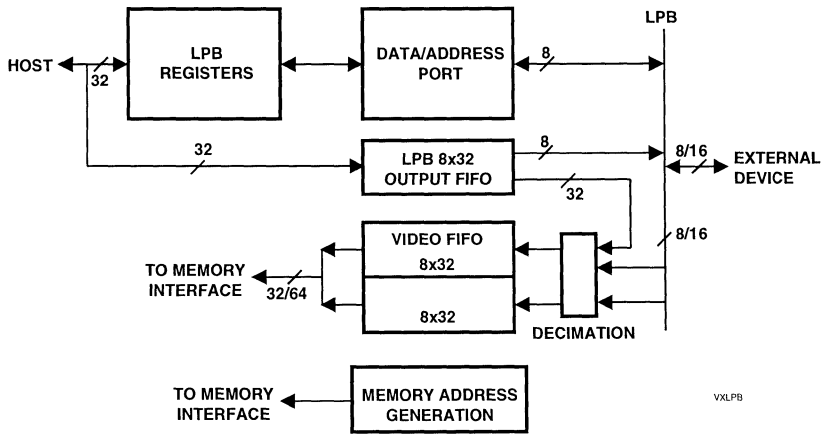


Figure 11-1. LPB Internal Block Diagram

11.1 Scenic/MX2 INTERFACE

The hardware interface to the Scenic/MX2 is shown in Figure 11-2.

The Scenic/MX2 interface is selected by setting MMFF00_3-1 to 000b. This interface is fully bi-directional. Scenic/MX2 registers can be accessed, compressed data sent and decompressed video data received.

11.1.1 Scenic/MX2 Register/Memory Access

To read/write a Scenic/MX2 register or private memory location (other than to transfer compressed data), the LPB Direct Read/Write Address register (MMFF14) is written. The new register/memory data is then written to MMFF18. For a write access, this write triggers the sequence shown in Figure 11-3 if the Scenic/MX2 is ready to receive the data ($\overline{\text{CREQ}}$/CRDY remains high). One cycle after ViRGE/VX asserts its $\overline{\text{VREQ}}$/VRDY signal, it sends the address in three byte writes. The first byte is composed of bits 23-16 of MMFF14. The three upper bits are 000b to define this as a write. Bit 4 is 1 for a register access and 0 for a memory access. Bits 3-0 are bits 19-16 of the address. The second byte is bits 15-8 of MMFF14 and the third byte is bits 7-0. The data immediately follows in four byte writes. Data is written in the opposite byte order to that for the address, i.e., least significant byte (bits 7-0) first and most significant byte (bits 31-24) last. ViRGE/VX then deasserts $\overline{\text{VREQ}}$/VRDY. The Host repeats the above sequence for another write if required.

If the Scenic/MX2 is not ready to receive data, it drives its $\overline{\text{CREQ}}$/CRDY signal low during the A0-0 byte (LSB) of the address phase. ViRGE/VX then delays sending the data until the Scenic/MX2 raises $\overline{\text{CREQ}}$/CRDY. This is depicted in Figure 11-4.

Figure 11-5 shows a Scenic/MX2 register/memory read when the Scenic/MX2 is ready to provide data. This is indicated by the Scenic/MX2 holding the $\overline{\text{CREQ}}$/CRDY high throughout the cycle. The three upper bits of the first address byte are 001 to define a read.

If the Scenic/MX2 is not ready to provide data, it drives its $\overline{\text{CREQ}}$/CRDY signal low during the address phase. ViRGE/VX then waits until the Sce-

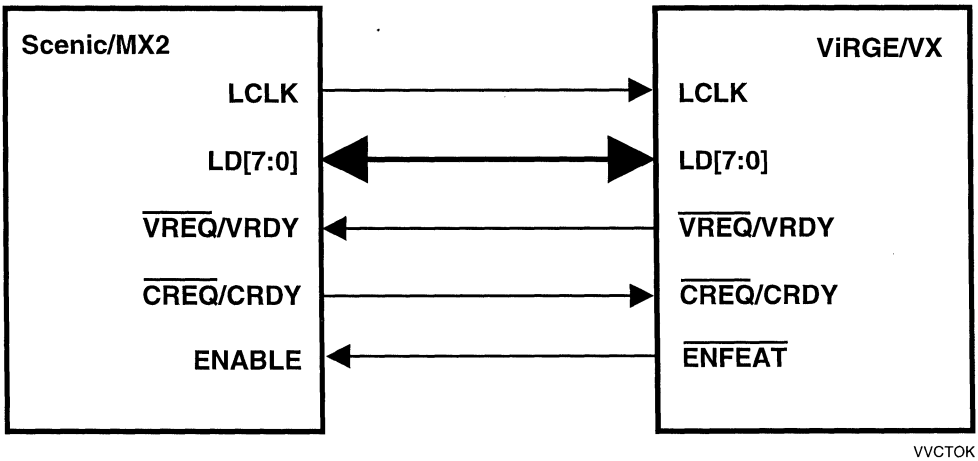


Figure 11-2. Scenic/MX2 Hardware Interface

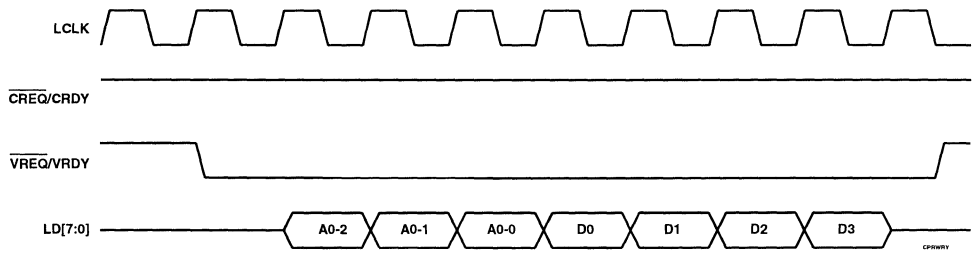


Figure 11-3. Scenic/MX2 Write (Scenic/MX2 Ready)

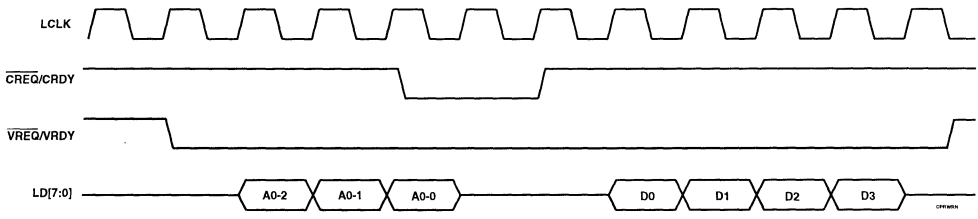


Figure 11-4. Scenic/MX2 Write (Scenic/MX2 Not Ready)

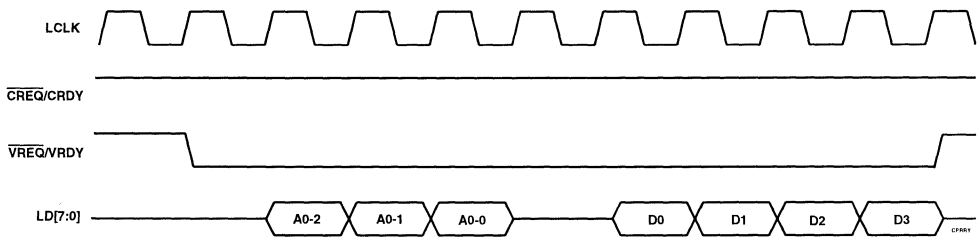


Figure 11-5. Scenic/MX2 Read (Scenic/MX2 Ready)

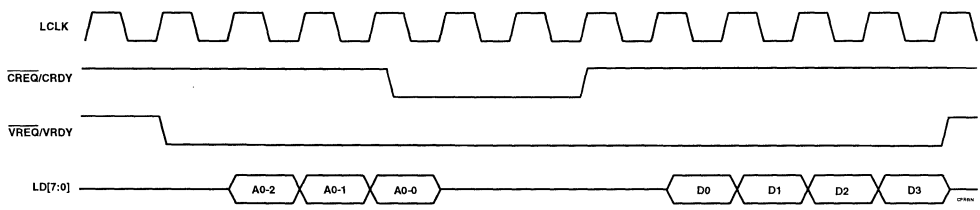


Figure 11-6. Scenic/MX2 Read (Scenic/MX2 Not Ready)

nic/MX2 raises $\overline{\text{CREQ}}$/CRDY and provides register data. This is depicted in Figure 11-6.

To prevent data starvation and deal with request contention, the following protocol is followed.

- No transaction can be initiated if the bus is active
- There is one dead cycle on the bus following all transactions
- One device may not initiate a transaction until the second cycle following the completion of a transaction initiated by the other device
- Neither device may initiate a transaction until the third cycle following the completion of a transaction initiated by itself
- If $\overline{\text{CREQ}}$/CRDY and $\overline{\text{VREQ}}$/VRDY are both driven low on the same cycle (request contention), $\overline{\text{CREQ}}$/CRDY (the Scenic/MX2) wins.

11.1.2 Scenic/MX2 Compressed Data Transfer

ViRGE/VX has an output FIFO for handling the transfer of compressed video data from the Host to the Scenic/MX2 (see Figure 11-1). The Host must first check the number of empty slots (MMFF04_3-0), then send no more than this many doublewords (32 bits) of compressed data to the FIFO. An eight doubleword address range (FF40H - FF5CH) is provided for this FIFO. Writes to any of these addresses are directed to the FIFO.

MMFF00_17-16 are programmed to specify the number of doublewords of data to burst to the Scenic/MX2. A write to the output FIFO then initiates a compressed data write to the Scenic/MX2. This is depicted in Figure 11-7 for a burst count of 2 (MMFF00_17-16 = 01b) for the case where the Scenic/MX2 is ready to receive the data. The address and first doubleword are transferred exactly as for a register/memory write. Following doublewords in the burst are each separated by one dead cycle. The address has no meaning except for the upper three bits, which are forced to 110b by hardware to specify a compressed data transfer. Note that burst writes that end because the FIFO is empty (as opposed to the maximum burst count being reached) hold $\overline{\text{VREQ}}$/VRDY low for one more cycle than is shown in Figure 11-7.

The Scenic/MX2 cannot accept a burst larger than eight doublewords. If MMFF00_17-16 are programmed to 11b (burst all) and eight doublewords are loaded into the FIFO, software must ensure that the FIFO is empty before loading more data into the FIFO.

A compressed data transfer when the Scenic/MX2 is not ready to receive data is almost the same as a register write for the same circumstances (see Figure 11-4). The only difference is that after the Scenic/MX2 returns its CRDY signal,

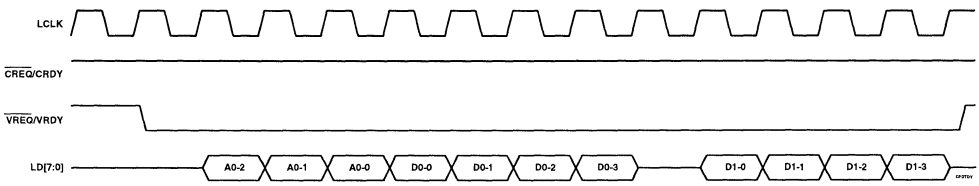


Figure 11-7. Compressed Data Xfer (Scenic/MX2 Ready)

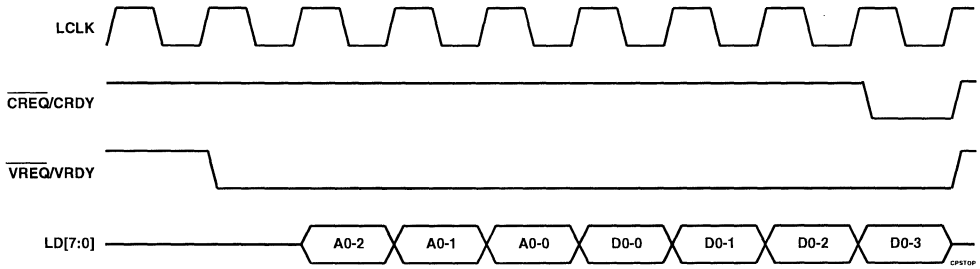


Figure 11-8. Scenic/MX2 Stopping a Compressed Xfer

additional doubleword packets may be burst to the Scenic/MX2 as shown in Figure 11-7.

The Scenic/MX2 can stop a compressed data transfer by pulling CREQ/CRDY low for one (and only one) cycle during byte three of any doubleword. This is shown in Figure 11-8.

An output FIFO empty interrupt can be enabled by setting MMFF08_17 to 1. The status is read via bit 1 of this same register.

11.1.3 Scenic/MX2 Video Capture

The following setup is done for Scenic/MX2 video capture:

- ViRGE/VX is placed in Scenic/MX2 mode (MMFF00_3-1 = 000b).
- One or two frame buffer starting addresses are defined (MMFF0C, MMFF10). One is required. The second is required for double buffering.
- The horizontal and vertical decimation registers are programmed (MMFF2C, MMFF30). This is optional.
- The line stride is programmed (MMFF34_10-0). This is not required if HSYNCs are not being sent.

ViRGE/VX signals its readiness to accept data by driving VREQ/VRDY high. This is done automatically when ViRGE/VX does not need to drive this

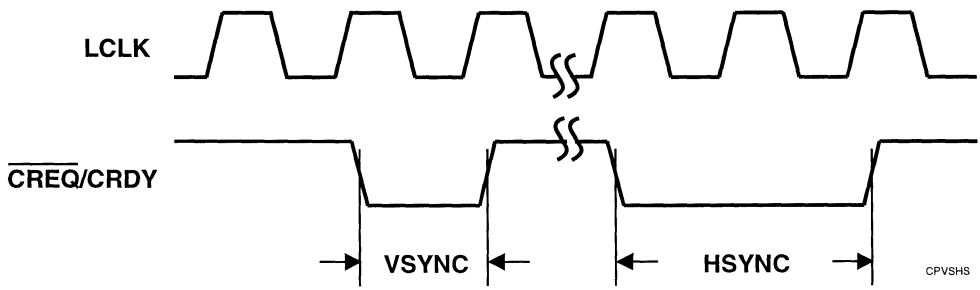


Figure 11-9. Scenic/MX2 VSYNC and HSYNC Protocols

signal low such as to initiate a register access or to indicate an LPB video FIFO full state. The Scenic/MX2 responds by sending a VSYNC ($\overline{\text{CREQ}}$/CRDY low for one cycle) followed by an HSYNC ($\overline{\text{CREQ}}$/CRDY low for two cycles). This is shown in Figure 11-9. As indicated in the figure, the time between VSYNC and HSYNC is variable. The HSYNC sequence occurs after each line, but may not occur before the first line, depending on how the Scenic/MX2 is programmed.

After the VSYNC/HSYNC sequence, the Scenic/MX2 can pull $\overline{\text{CREQ}}$/CRDY low at any time and begin sending data three clocks later. This is shown in Figure 11-10. ViRGE/VX assumes data has begun any time $\overline{\text{CREQ}}$/CRDY is held low for more than two cycles. When the Scenic/MX2 is sending the last byte, it drives $\overline{\text{CREQ}}$/CRDY high. The Scenic/MX2 must always send data in 4-byte packets. If it has fewer to send for the last packet, it must pad the transmission with dummy writes to create a 4-byte packet.

Figure 11-10 shows what happens when ViRGE/VX is ready to receive all the data. If ViRGE/VX cannot accept more data, such as when its LPB video FIFO is full, it drives its $\overline{\text{VREQ}}$/VRDY signal low during the first byte

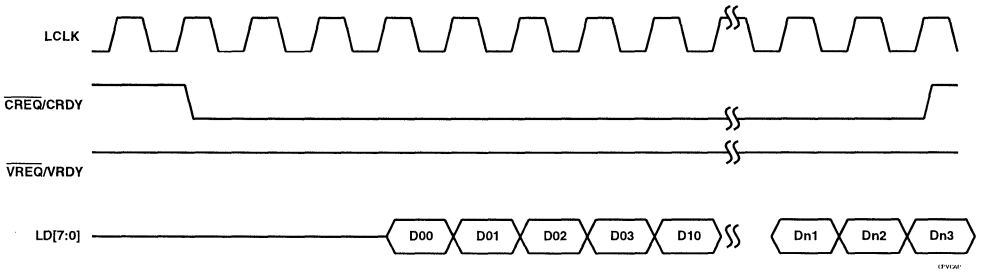


Figure 11-10. Scenic/MX2 Video Input (ViRGE/VX Ready)

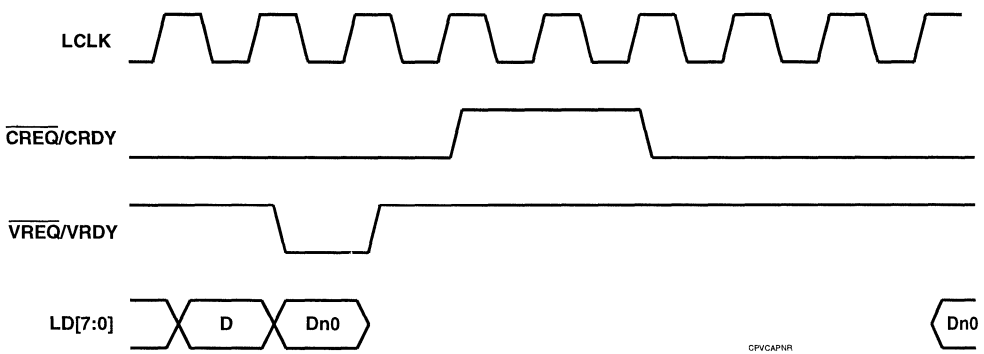


Figure 11-11. Scenic/MX2 Video Input (ViRGE/VX Not RDY

phase of a 4-byte packet. All bytes starting with this one are rejected by ViRGE/VX and must be resent by the Scenic/MX2 after ViRGE/VX drives its $\overline{\text{VREQ}}$/VRDY signal high again. This is depicted in Figure 11-11, where the Dn0 byte, which is the first byte of the nth 4-byte packet, is rejected. When ViRGE/VX can accept more data, it drives $\overline{\text{VREQ}}$/VRDY high. The Scenic/MX2 drives $\overline{\text{CREQ}}$/CRDY high (two cycles later) and then drives it low when it is ready to resend the data. The resend of Dn0 and subsequent bytes starts two cycles later.

When ViRGE/VX receives an HSYNC from the Scenic/MX2, it adds the line offset (MMFF34_10-0) to the previous line starting address and starts writing the next data at that location. In this way, for example, it can transfer 640-byte lines into a frame buffer configured for 1024-byte lines. If HSYNCs are not sent, memory will be written in a contiguous manner.

11.2 DIGITIZER INTERFACE

The hardware interface to the Philips digitizer in Video 8 In mode (MMFF00_3-1 = 010b) is shown in Figure 11-12. This section describes the interface to the Philips SAA7110 digitizer.

The functional timing for converting the SAA7110 16-bit video output to the 8-bit input required by the LPB in a VL-Bus configuration is shown in Figure 11-13.

In Video 16 mode (MMFF00_3-1 = 001b), which is available only for PCI configurations, no data conversion is required. Y[7:0] connect to LD[7:0] and UV[7:0] connect to LD[15:8]. LLC2 connects to LCLK and LLC is not connected.

As an alternative, the Scenic/MX2 provides a glueless interface to the SAA7110. In this case, the Scenic/MX2 handles the 16-bit to 8-bit conversion and also provides the I^2C interface to the SAA7110. ViRGE/VX then receives the video data, clock and controls from the Scenic/MX2. The Scenic/MX2 documentation describes this interface.

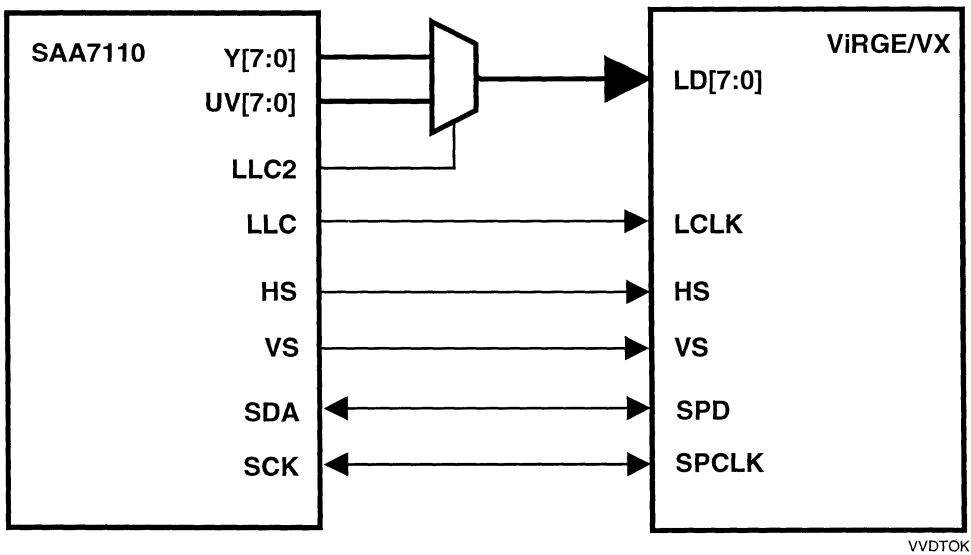


Figure 11-12. SAA7110 Digitizer Interface

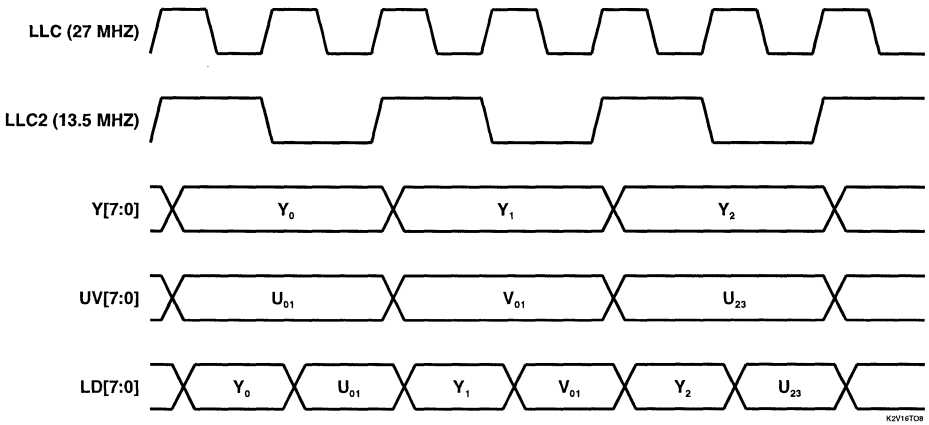


Figure 11-13. 16- to 8-bit Video Data Conversion

11.2.1 I²C Register Interface

SAA7110 registers are programmed via a serial I²C interface. This interface is described in Section 12.

11.2.2 SAA7110 Video Input

The following setup is done for SAA7110 video input:

- ViRGE/VX is placed in Video 8 In mode (MMFF00_3-1 = 010) or Video 16 mode (MMFF00_3-1 = 001b) for PCI configurations.
- Byte swapping is disabled by setting MMFF00_6 to 1.
- The correct vertical and horizontal sync polarities are specified (MMFF00_9, 10).
- One or two frame buffer starting addresses are defined (MMFF0C, MMFF10). One is required. The second is required for double buffering.

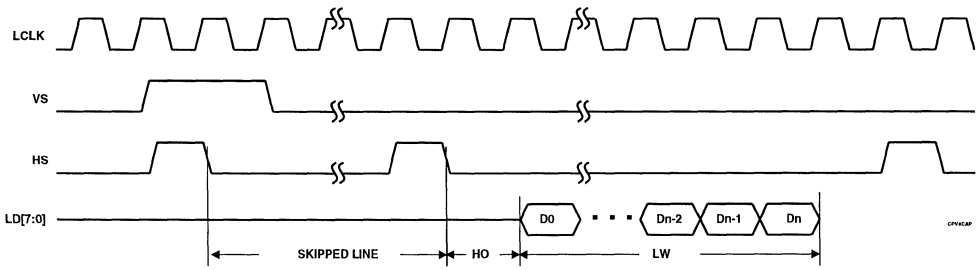


Figure 11-14. Video 8 In or 16 Mode Input

- The horizontal and vertical decimation registers are programmed (MMFF2C, MMFF30). This is optional.
- The video input window size (height in lines and width in pixels) is programmed in MMFF24.
- The video data horizontal and vertical offsets are programmed in MMFF28.
- The line offset (stride) is programmed (MMFF34_10-0).

The SAA7110 then sends video data as shown in Figure 11-14. In this figure, both VSYNC (VS) and HSYNC (HS) have active high polarity. The vertical offset (MMFF28_24-16) is 1, meaning the first line is skipped. The horizontal offset HO (MMFF28_11-0) is 1, meaning that the first data starts one clock after the second HS goes low. HS goes high again some time after the last byte of the line, whose position is specified by the line width (LW) programmed in MMFF24_11-0. The widths of the VS and HS pulses shown may vary.

Alternate frames of the video input can be discarded (not written to memory) by setting bit 5 of MMFF00 to 1.

11.3 HOST PASS-THROUGH

When pass-through mode is enabled (MMFF00_3-1 = 100b), the CPU can write 32-bit data to the output FIFO and have this data passed directly to the decimation block (bypassing the LPB bus). The data are sent exactly as for compressed video data to an MPEG decoder. The data will then be decimated according to the programming of MMFF2C (horizontal) and MMFF30 (vertical) and then passed to the video FIFO to be written to display memory. This path is shown in Figure 11-1.

When the Host sends an HSYNC (MMFF00_12 = 1) or VSYNC (MMFF00_11), the decimation registers are re-loaded. Therefore, the Host must ensure that at least 5 clocks pass between the sync and the start of data to allow time for this reloading.

When pass-through is used in LPB mode, bit 24 of MMFF00 provides the option of using SCLK to clock the LPB function.

Pass-through is not supported if big-endian addressing is being used.

11.4 LPB-ENABLED PIN ASSIGNMENTS

The pin assignments when the various LPB modes are enabled are shown in Table 11-1. Note that some functions are available only in PCI configurations. These have (PCI) next to the pin number.

Table 11-1. LPB-Enabled Pin Assignments

Pin #	Scenic/MX2 MMFF00_3-1 = 000	Video 16 or 8 In MMFF00_3-1 = 001 MMFF00_3-1 = 010
A13	LD0	LD0
D13	LD1	LD1
B13	LD2	LD2
A14	LD3	LD3
D15	LD4	LD4
B14	LD5	LD5
C16	LD6	LD6
A15	LD7	LD7
C11 (PCI)	NO FUNCTION	LD8 (Video 16)
A10 (PCI)	NO FUNCTION	LD9 (Video 16)
B9 (PCI)	NO FUNCTION	LD10 (Video 16)
C10 (PCI)	NO FUNCTION	LD11 (Video 16)
A9 (PCI)	NO FUNCTION	LD12 (Video 16)
D7 (PCI)	NO FUNCTION	LD13 (Video 16)
A6 (PCI)	NO FUNCTION	LD14 (Video 16)
C7 (PCI)	NO FUNCTION	LD15 (Video 16)
D17	LCLK	LCLK
B5	$\overline{\text{VREQ}}$/VRDY	HS
A5	$\overline{\text{CREQ}}$/CRDY	VS

Section 12: Miscellaneous Functions

This section explains how ViRGE/VX interfaces to the video BIOS ROM and feature connector. Green PC support, the General I/O Ports, the serial communications port and interrupt generation are also described.

12.1 VIDEO BIOS ROM INTERFACE

The video BIOS ROM contains power-on initialization, mode setup, and video data read/write routines. The video BIOS can be part of the system ROM or it can be implemented separately.

A separate implementation of the video BIOS is shown in Figure 12-1. The RD[7:0] and RA[15:0] signals are multiplexed on PD pins. Therefore, the BIOS ROM must be shadowed immediately after reset and BIOS access disabled to prevent interference with graphics operation.

Figure 12-2 depicts the PCI configuration functional timing for reading one byte from the ROM. $\overline{\text{ROMEN}}$ is asserted to drive the byte of read data at the address on RA[15:0]. ViRGE/VX latches the data one clock before deassertion of $\overline{\text{ROMEN}}$ and then drives this data onto the AD bus.

ViRGE/VX also supports 16- and 32-bit ROM reads, as defined by the states of the byte enables. For a 16-bit read, ViRGE/VX automatically increments the lower address once and generates the second byte of read data. For a 32-bit read, ViRGE/VX automatically increments the lower address three times and generates the remaining three bytes of read data. In both cases, $\overline{\text{TRDY}}$ is delayed until all the required data is

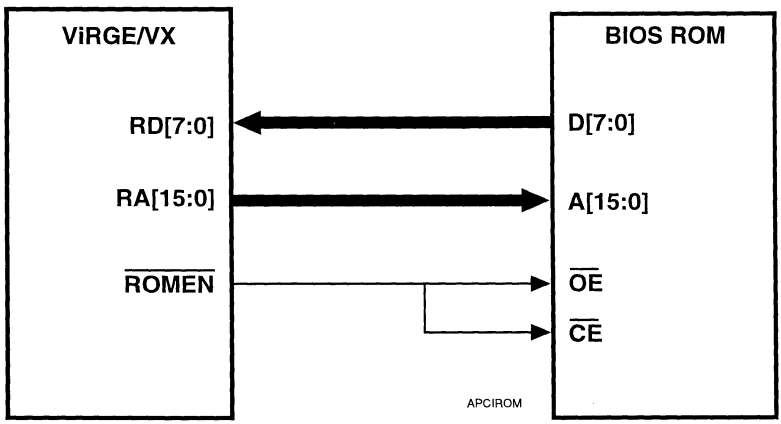


Figure 12-1. BIOS ROM Interface

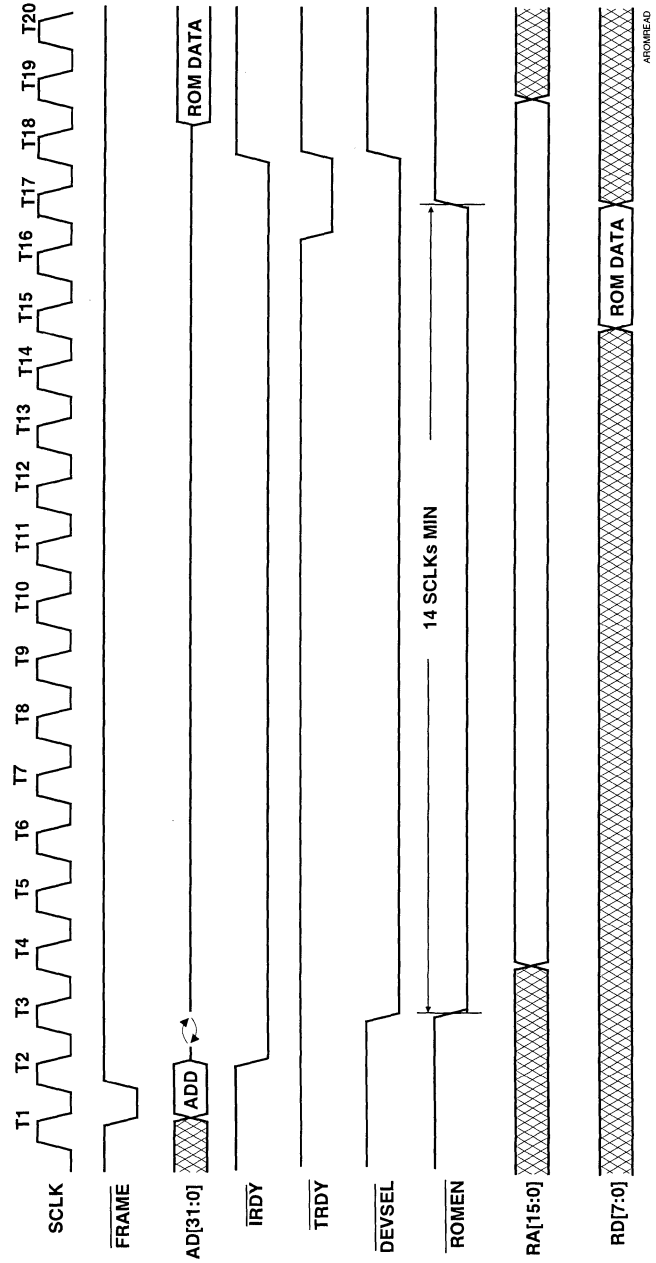


Figure 12-2. BIOS ROM Read Functional Timing

available on the AD bus. For 16-, 24- or 32- bit accesses, the ROM access time must be 10 SCLKs or less, as opposed to the 14 SCLKs shown in Figure 12-3 for an 8-bit access.

ViRGE/VX maps the CPU memory address spaces for the video BIOS ROM into physical ROM addresses. PCI systems support a relocatable 64-KByte video BIOS address range via the BIOS ROM Base Address configuration register (Index 30H).

Bit 0 of the BIOS ROM Base Address register (Index 30H) is cleared to 0 to disable BIOS accesses.

12.2 GREEN PC SUPPORT

ViRGE/VX provides support for the VESA Display Power Management Signaling (DPMS) protocol by allowing independent control of the HSYNC and VSYNC signals. To use this capability, the bit pattern xxxx0110b must be written to the SR8 register to unlock access to the SRD register. Bits 5-4 of SRD then control the state of HSYNC and bits 7-6 of SRD control the state of VSYNC.

Driving pin 165 ($\overline{\text{PDOWN}}$) low turns off the RGB analog outputs of the internal DACs.

12.3 GENERAL INPUT PORT

ViRGE/VX provides a 4-bit General Input Port (GIP) as part of its LPB function. The following steps are required to implement it.

1. Disable all other LPB uses.
2. Enable sensing of the desired input data on LD[7:4].
3. If the LPB General Output Port function is also in use, ensure that the correct output data is programmed in MMFF1C_3-0.
4. Program SR1C_1 to 1 to select $\overline{\text{STWR}}$ on pin B8.
5. Write (anything) to CR5C. The data on LD[7:4] are latched 2 DCLKs later into MMFF1C_7-4. (This also drives the contents of MMFF1C_3-0 onto LD[3:0] and generates the $\overline{\text{STWR}}$ pulse on pin B8. The input data is latched on the rising edge of $\overline{\text{STWR}}$. See Figure 12-3)
6. Disable sensing of input data on LD[7:4].

12.4 GENERAL OUTPUT PORT

ViRGE/VX provides a 4-bit General Output Port (GOP) as part of its LPB function. To implement this:

1. Disable all other LPB uses.
2. Program the desired output in MMFF1C_3-0.

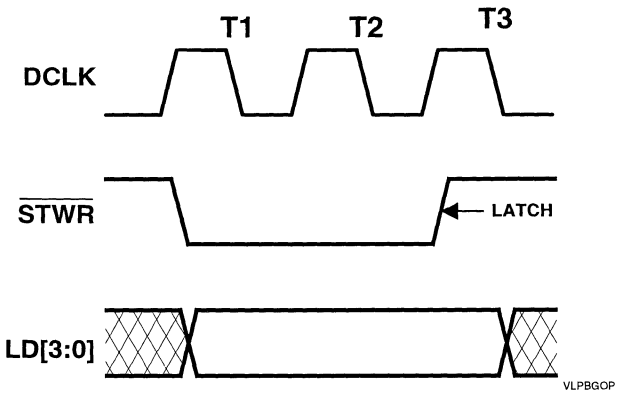


Figure 12-3. General I/O Port Timing

4. Program SR1C_1 to 1 to enable output of $\overline{\text{STWR}}$ on pin B8.
5. Write (anything) to CR5C. The data in MMFF1C_3-0 are immediately driven onto LD[3:0] and the $\overline{\text{STWR}}$ pulse is generated. The rising edge of $\overline{\text{STWR}}$ (2 DCLKs after it is asserted) can be used to latch the data into an external device. The data is held valid for 1/2 DCLK after this edge. See Figure 12-3.

ViRGE/VX also provides a 2-bit GOP on dedicated pins. To implement this:

1. Set SR1C_1 to 1.
2. Program the desired output in CR5C_1-0. This statically drives the state of CR5C_0 onto pin A16 and the state of CR5C_1 onto pin B8. These pins will continue to reflect the register bit states as long as SR1C_1 =1. The values in CR5C_1-0 can be reprogrammed at any time.

Pins A16 and B8 are driven high on power-up. Thus, external devices with active low enables will not be enabled when connected to these pins.

12.5 FEATURE CONNECTOR INTERFACE

Setting SRD_1 to 1 selects LPB feature connector operation. This configuration provides an interface to either a baseline VESA Advanced Feature Connector (VAFC) or pass-through bidirectional feature connector. In all cases, SRD_0 must be set to 1 to enable feature connector operation and SR1C_1-0 must be 00b to enable $\overline{\text{ENFEAT}}$ on pin A16. In addition, LPB operation must be disabled, (MMFF00_0 = 0) and Streams Processor operation must be disabled (CR67_3-2 = 00b) before feature connector operation is enabled.

ViRGE/VX supports a 16-bit bi-directional feature connector (VAFC). The pins used to provide this type of operation are listed in Table 12-1. The interface is shown in Figure 12-4.

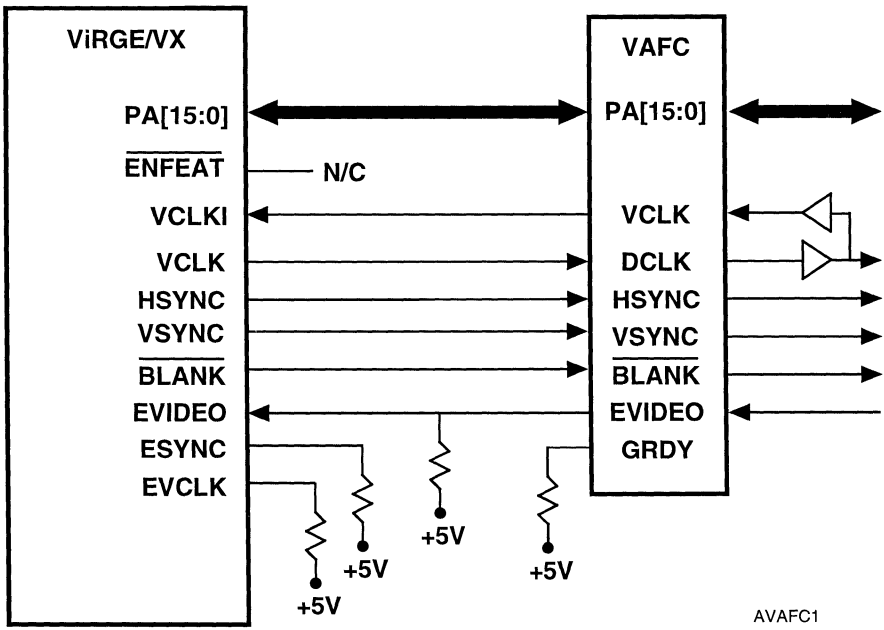


Figure 12-4. VAFC Implementation

Table 12-1 LPB Feature Connector Configuration

Pin(s)	Signals
C7, A6, D7, A9, C10, B9, A10, C11, A15, C16, B14, D15, A14, B13, D13, A13	PA[15:0]
A16	$\overline{\text{ENFEAT}}$
A8	$\overline{\text{BLANK}}$
D17	VCLK
A7	VCLKI
B10	ESYNC
B5	EVIDEO
A5	EVCLK
B16	HSYNC
C17	VSYNC

ViRGE/VX also supports a standard 8-bit pass-through feature connector as shown in Figure 12-5.

12.6 SERIAL COMMUNICATIONS PORT

A serial communications port is implemented in the MMFF20 register. Bit 4 is set to 1 to enable the interface. The clock is written to bit 0 (= 0) and data to bit 1 (= 0), driving the SPCLK and SPD pins low respectively. The state of the SPCLK pin can be read via bit 2 and the state of the SPD pin can be read via bit 3. The SPCLK and SPD pins are tri-stated when their corresponding control bits are reset to 0, allowing other devices to drive the serial bus.

Typical uses for the serial port are for DDC monitor communications and I^2C interfacing. When SPCLK and SPD are tri-stated, ViRGE/VX can detect an I^2C start condition (SPD driven low while SPCLK is not driven low). This condition is generated by another I^2C master that wants control of the I^2C bus. If bit 19 of MMFF08 is set to 1, detection of a start condition generates an interrupt and sets bit 3 of MMFF08 to 1. If bit 24 of MMFF08 is set to 1, ViRGE/VX drives SPCLK low to generate I^2C wait states until the Host can clear the interrupt and service the I^2C bus.

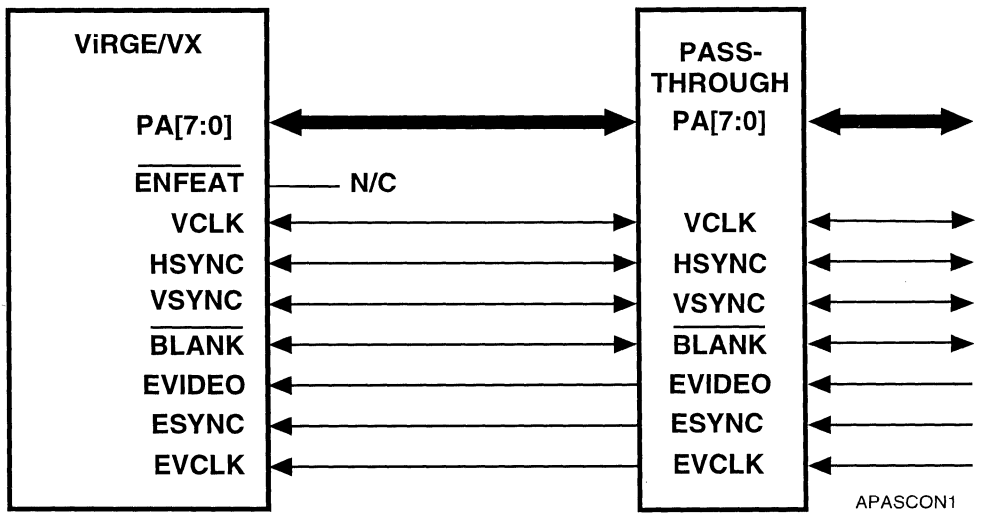


Figure 12-5. Pass-Thru Feature Connector

If PD26 is strapped low at reset, strapping of PD25 selects either E2H (PD25 pulled high) or E8H (PD25 pulled low) as the I/O port address for the serial port register MMFF20. This allows the ports to be used for serial communications, typically I^2C, when ViRGE/VX is not enabled. If analog switches are used for isolation as explained in the previous paragraph, designers must ensure that the I^2C function is enabled by default on reset. If I/O access is desired after ViRGE/VX has been enabled and then disabled, programmers must ensure that the I^2C function is selected before ViRGE/VX is disabled because the General Output Port may not be available to change the selection.

12.7 INTERRUPT GENERATION

The $\overline{\text{INTA}}$ pin is pulled low to signal an interrupt. Whatever the mode of operation (VGA or Enhanced), bit 4 of CR32 must be set to 1 to enable interrupt generation.

When ViRGE/VX is being operated in VGA mode (CR66_0 = 0), only a vertical retrace can generate an interrupt. This is enabled when bit 5 of CR11 is cleared to 0 and a 1 has been programmed into bit 4 of CR11. When an interrupt occurs, it is cleared by writing a 0 to bit 4 of CR11. The interrupt must then be re-enabled by writing a 1 to the same bit. Note that the BIOS clears both bit 4 and bit 5 of CR11 to 0 during power-on, a mode set or a reset. Thus, interrupt generation is disabled until bit 4 is set to 1.

When ViRGE/VX is being operated in Enhanced mode (CR66_0 = 1), interrupts can be generated by a vertical retrace, S3d Engine busy, S3d Engine done, Host DMA done, Command DMA done, S3d FIFO empty, command FIFO overflow and command FIFO empty. These interrupts are enabled and cleared and their status reported via MM8504.

Multiple interrupts can be enabled at the same time in Enhanced mode. The interrupt pin will remain asserted until all interrupt status bits are cleared.

Section 13: Basic Software Functions

This section describes the basic operations required for ViRGE/VX.

13.1 CHIP WAKEUP

The following code segment wakes up ViRGE/VX.

```
mov dx,3c3h        ; Video Subsystem Enable register address
mov al,01h         ; bit 0 = 1, enable graphics display
out dx,al          ; write new bit values to 3c3h
mov dx,3cch        ; Miscellaneous Output Read register
in al,dx           ; Read 3cch
[load CRTCs]       ; program CRTC registers
mov dx,3C6h        ; DAC Mask register address
mov al,FFh         ; DAC Mask register initialization value
out dx,al          ; Initialize DAC mask and release BLANK signal
   .
   .
   .
```

13.2 REGISTER ACCESS

S3 has added a number of graphics registers to the standard VGA set. These can be locked when not in use to prevent accidental access and unlocked when access is requires. This section explains how this is done.

13.2.1 Unlocking the S3 Registers

The S3 registers (CR30 and higher plus the Enhanced Commands registers) must be unlocked before they can be accessed by the CPU. The code to do this is:

Note: Byte operations are used in the following examples for clarity. Word operations, e.g..,

mov ax, 4838h
out dx,ax

should be used for efficiency instead of the operations used in the first example below.

```
; Write code to SR8 to provide access to the S3 extended Sequencer registers
(SR9-SRFF)
;
   mov dx,3c4h        ; copy index register address into dx
   mov al,08h         ; copy index for SR8 register into al
   out dx,al          ; write index to index register
   inc dx             ; increment dx to 3c5h (data register address)
   mov al,06h         ; copy unlocking code (xxxx0110b, x=don't care) to al
   out dx,al          ; write the unlocking code to the data register
   dec dx             ; restore the index register address to dx
;
; Write code to CR38 to provide access to extended CRTC registers CR2D-CR3F
;
   mov dx,3d4h        ; copy index register address into dx
   mov al,38h         ; copy index for CR38 register into al
   out dx,al          ; write index to index register
   inc dx             ; increment dx to 3D5h (data register address)
   mov al,48h         ; copy unlocking code (01xx10xxb, x=don't care) to al
   out dx,al          ; write the unlocking code to the data register
   dec dx             ; restore the index register address to dx
;
; Write code to CR39 to provide access to extended CRTC registers CR40-CRFF
;
; dx is already loaded with 3D4h because of the previous instruction
;
   mov al,39h         ; copy index for CR39 register into al
   out dx,al          ; write index to index register
   inc dx             ; increment dx to 3D5h (data register address)
   mov al,0a5h        ; copy unlocking code to al (the code a5H also unlocks
                      ; access to configuration registers CR36, CR37 and CR68
   out dx,al          ; write the unlocking code to the data register
   dec dx             ; restore the index register address to dx
;
; Set bit 0 in CR40 to enable access to the Enhanced Programming registers.
;
; dx is already loaded with 3D4h because of previous instruction
   mov al,40h         ; copy index for CR40 register into al
   out dx,al          ; write index to index register
   inc dx             ; increment dx to 3D5h (data register address)
   in al,dx           ; read register data for read/modify/write operation
   or al,1            ; set bit 0 to 1
   out dx,al          ; write the unlocking code to the data register
   dec dx             ; restore the index register address to dx
```

13.2.2 Locking the S3 Registers

Relocking the S3 registers is done by repeating the code used to unlock the registers except:

1. The values written to the SR8, CR38 and CR39 registers must change at least one of the significant bits in the valid code pattern. For example, 00h will always accomplish this.

2. After first verifying that the S3D Engine is not busy (bit 9 of 9AE8H is 0), bit 0 of CR40 must be cleared to 0. A read-modify-write cycle must be used instead of the code used above to prevent overwriting of any changes made to bits 7-1 in CR40 since reset.

```
mov dx,3d4h        ; copy index register address into dx
mov al,40h         ; copy index for CR40 register into al
out dx,al          ; write index to index register
inc dx             ; increment dx to 3D5h (data register address)
in al,dx           ; read content of CR40 into al
and al,0feh        ; clear bit 0 to 0
out dx,al          ; write to CR40 to lock the Enhanced Commands registers
dec dx             ; restore the index register address to dx
```

13.3 TESTING FOR THE PRESENCE OF A ViRGE/VX CHIP

After unlocking, an ViRGE/VX chip can be identified via CR2E.

```
   mov dx,3d4h        ; copy index register address into dx
   mov al,2eh         ; copy index for CR2E register into al
   out dx,al          ; write index to index register
   inc dx             ; increment dx to 3D5h (data register address)
   in al,dx           ; read content of CR2E into al
   cmp al,3dh         : compare chip ID to the desired chip ID (3dh)
   jne not_VX         ; jump to not_ViRGE/VX if ID for ViRGE/VX is not found
   .                  ; ViRGE/VX chip found - continue
   .
   .
```

13.4 GRAPHICS MODE SETUP

Some programs may require a graphics mode other than that provided by standard operation. For example, a DOS game may require a resolution of 640x400x8 (VESA mode 100) instead of the standard DOS mode, e.g., mode 03. The following code fragment shows how this is done.

```
mov ax,4f02h       ; VESA super VGA mode function call
mov bx,100h        ; mode 100
int 10h            ; call video BIOS
```

Section 14: VGA Compatibility Support

This section describes ViRGE/VX support for standard VGA and VESA Super VGA graphics standards.

14.1 VGA COMPATIBILITY

ViRGE/VX is compatible with the VGA standard. These modes are not accelerated using the S3d Engine. However, other design features provide excellent VGA performance.

Several of the standard VGA registers have been modified or extended in ViRGE/VX. Table 14-1 describes these changes.

Table 14-1. Standard VGA Registers Modified or Extended in ViRGE/VX

Register	Change to Standard VGA Definition
CR0	Extension bit 8 is bit 0 of CR5D. Bit 5 of CR35 controls access to this register. Bit 7 of CR43 doubles the parameter size.
CR1	Extension bit 8 is bit 1 of CR5D. Bit 5 of CR35 controls access to this register. Bit 7 of CR43 doubles the parameter size.
CR2	Extension bit 8 is bit 2 of CR5D. Bit 5 of CR35 controls access to this register. Bit 7 of CR43 doubles the parameter size.
CR3	The length of the blanking pulse defined in this register can be extended by 64 DCLKs via bit 3 of CR5D. Bit 5 of CR35 controls access to this register. Bit 7 of CR43 doubles the parameter size.
CR4	Extension bit 8 is bit 4 of CR5D. Bit 5 of CR35 controls access to this register. Bit 7 of CR43 doubles the parameter size.
CR5	The length of the HSYNC pulse defined in this register can be extended by 32 DCLKs via bit 5 of CR5D. Bit 5 of CR35 controls access to this register. Bit 7 of CR43 doubles the parameter size.
CR6	In addition to the standard VGA extensions (bit 8 is bit 0 of CR7, bit 9 is bit 5 of CR47), bit 10 is bit 0 of CR5E. Bit 4 of CR35 controls access to this register.
CR7	Bit 4 of CR35 controls access to bits 0, 2, 3, 5 and 7 of this register.
CR9	Bit 4 of CR35 controls access to bit 5 of this register.
CRC	The display start address is a 21-bit value for ViRGE/VX. The extension bits (20-16) are bits 4-0 of CR69.
CRE	The cursor location address is a 21-bit value for ViRGE/VX. The extension bits (20-16) are bits 4-0 of CR69.

CR10	In addition to the standard VGA extensions (bit 8 is bit 2 of CR7, bit 9 is bit 7 of CR7), bit 10 is bit 4 of CR5E. Bit 4 of CR35 controls access to this register.
CR11	Bit 4 of CR35 controls access to bits 3-0 of this register. Bit 6 (3/5 refresh cycles per line) can be overridden by CR3A_2-0. Setting bit 1 of CR33 to 1 disables the write protect effect of bit 7 of this register on bits 1 and 6 of CR7.
CR12	In addition to the standard VGA extensions (bit 8 is bit 1 of CR7, bit 9 is bit 6 of CR7), bit 10 is bit 1 of CR5E.
CR13	Bit 2 of CR43 is the old extension bit (bit 8) of this register. Bits 5-4 of CR51 are the new extension bits (bits 9-8) of this register.
CR15	In addition to the standard VGA extensions (bit 8 is bit 3 of CR7, bit 9 is bit 5 of CR9), bit 10 is bit 2 of CR5E. Bit 4 of CR35 controls access to this register.
CR16	Bit 4 of CR35 controls access to this register.
CR17	Bit 5 of CR35 controls access to bit 2 of this register.
CR18	In addition to the standard VGA extensions (bit 8 is bit 4 of CR7, bit 9 is bit 6 of CR9), bit 10 is bit 6 of CR5E.
AR00-AR0F	Bit 6 of CR33 controls access to these registers.
3C6H-3C9H	Bit 4 of CR33 controls writes to these registers.

For a detailed discussion of VGA programming, see *Programmer's Guide to the EGA, VGA and Super VGA Cards, 3rd Edition* by Richard F. Ferraro (Addison-Wesley Publishing Company, Inc).

14.2 VESA SUPER VGA SUPPORT

ViRGE/VX supports the extended (Super) VGA modes defined by VESA. All modes are accelerated by the S3d Engine except for the planar (4 bits/pixel) ones.

Section 15: Enhanced Programming

Enhanced mode provides a level of performance far beyond what is possible with the VGA architecture. Hardware BitBLTs (with 256 ROPs), 2D and 3D line drawing, 2D polygon fills and 3D triangle drawing are implemented. Hardware cursor support and clipping are also supported. While in Enhanced mode, the display memory bit map can be updated in two ways. One is to have the CPU issue commands and send data to the S3d Engine, which then controls pixel updating. The other is to have the CPU write directly to memory. (This is also possible in non-Enhanced modes via paging.) This section explains these two methods and provides a set of Enhanced mode 2D programming examples and explains the basic elements of 3D drawing.

15.1 MEMORY-MAPPED I/O

ViRGE/VX provides two memory-mapped I/O (MMIO) methods. For the "old" method, the base address is A000H (or B800H), allowing use during DOS and real mode operation. For the "new" method, the base address is the linear addressing (or PCI) base address and requires protected mode. In addition, address space is provided for linear addressing and big endian addressing. Each of these MMIO methods is described below.

15.1.1 Old MMIO

Setting bits 4-3 of CR53 to 10b enables the old MMIO function. A setting of 11b enables both the old and new MMIO methods simultaneously. When the old MMIO is enabled, CR53_5 selects the base address. CR53_5 = 0 places the MMIO window at A0000H - AFFFFH. CR53_5 =1 places the MMIO window at B8000H - BFFFFH. The latter setting leaves A0000H - B7FFFH free for VGA memory and other uses. In either case, all the ViRGE/VX registers are accessible via either window at the variable offsets shown in Table 15-1. For example, the PCI configuration space registers are found starting at A8000H (or B8000H, depending on the setting of CR53_5).

With old MMIO enabled and CR53_5 =0, image writes are made by accessing any memory location in the 32-KByte address space from A0000H to A7FFFH. This allows efficient use of the MOVSW and MOVSD assembly language commands. Accesses must be to doubleword addresses. Software must not make image writes beyond the A7FFFH range. If CR53_5 = 1, image writes cannot be made as the A0000H - A7FFFH range is reserved.

When MMIO is enabled (old or new), clearing bit 7 of SR9 to 0 allows both programmed I/O (IN, OUT) access and MMIO (MOV) access. Setting this bit to 1 disables programmed I/O access, allowing only MMIO access. The latter is required for plug and play operation.

15.1.2 New MMIO

The new MMIO method for ViRGE/VX provides a 64-MByte addressing window starting at the base address specified in CR59-5A or the PCI base address register. This space is divided into a 32-MByte space for little endian (Intel-style) addressing and a 32-MByte space for big endian (Power PC-style) addressing. All registers and data transfer locations are mapped into this area as shown in Table 7-1.

The new MMIO (only) is enabled by setting bits 4-3 of CR53 to 01b. This is the default for a PCI bus configuration, allowing PCI software immediate access to all registers and the ability to relocate the address space. The new MMIO is also enabled in conjunction with the old MMIO method when bits 4-3 of CR53 are set to 11b.

When MMIO is enabled (old or new), clearing bit 7 of SR9 to 0 allows both programmed I/O (IN, OUT) access and MMIO (MOV) access. Setting this bit to 1 disables programmed I/O access, allowing only MMIO access. The latter is required for plug and play operation.

Table 15-1. New MMIO Addresses

Lower 32 MBytes - Little Endian Addressing	
Description	**Offset From Base (Hex)**
Linear Addressing (16M)	000 0000 - 0FF FFFF
Image Data Transfer (32K)	100 0000 - 100 7FFF
PCI Configuration Space Registers	100 8000 - 100 8043
Streams Processor Registers	100 8180 - 100 81FF
Memory Port Controller	100 8200 - 100 8224
CRT VGA 3B? Registers	100 83B0 - 100 83Bx
CRT VGA 3C? Registers	100 83C0 - 100 83Cx
CRT VGA 3D? Registers	100 83D0 - 100 83Dx
Subsystem Status Enhanced Register	100 8504
Advanced Function Control Register	100 850C
DMA Controller Registers	100 8580 - 100 85FF
Color Pattern Registers	100 A000 - 100 A1FF
BitBLT/Rectangle Fill Registers	100 A400 - 100 A5FF
2D Line Draw Registers	100 A800 - 100 A9FF
2D Polygon Fill Registers	100 AC00 - 100 ADFF
3D Line Draw Registers	100 B000 - 100 B1FF
3D Triangle Registers	100 B400 - 100 B5FF
Local Peripheral Bus Registers	100 FF00 - 100 FF5C

Values in the gaps between the memory ranges shown in Table 15-1 are reserved.

For big endian addressing, add 2 to the most significant hex digit shown in Table 15-1, i.e., 0xx xxxx becomes 2xx xxxx and 1xx xxxx becomes 3xx xxxx. Thus, the total address space decoded by ViRGE/VX is 64 MBytes.

15.2 DIRECT BITMAP ACCESSING—LINEAR ADDRESSING

Linear addressing is useful when software requires direct access to display memory. ViRGE/VX provides two linear addressing schemes. The old method can be used when MMIO is disabled or with the old MMIO method. The second is used in conjunction with the new MMIO method.

15.2.1 Old Linear Addressing

Enhanced mode operation must be enabled before linear addressing is enabled. This means that bit 0 of CR63 is set to 1 to enable Enhanced mode functions and bit 3 of CR31 is set to 1 to specify Enhanced mode memory mapping.

ViRGE/VX provides linear addressing of up to 8 MBytes of display memory. Linear addressing of more than 64 KBytes requires that the CPU be operated in protected mode.

The S3d Engine busy flag, bit 13 of MM8504 (read), should be verified to be 0 (not busy) before linear addressing is enabled by setting bit 4 of CR58 to 1. The size of the linear address window is set via bits 1-0 of CR58. The base address for the linear addressing window is set via CR59 and CR5A (or via the Base Address 0 (Index 10H) PCI configuration register for PCI systems).

For operation in real mode, the linear addressing window size can be set to 64 KBytes. The base address for the window is set to A0000H by programming bits 31-16 of the window position in CR59-CR5A to 000AH. If bit 0 of CR31 is set to 1, the memory page offset (64K bank) specified in bits 5-0 of CR6A is added to the linear addressing window position base address, allowing access to up to 4 MBytes of display memory through a 64-KByte window.

15.2.2 New Linear Addressing

With the new MMIO enabled (CR53_4-3 = 01b or 11b), the first 16 MBytes of each 32M address space (big and little endian) are dedicated to linear addressing. A maximum of 8 MBytes of each address space (starting at the lowest address of the space) is usable with ViRGE/VX. The base address is taken from bits 31-26 of the linear address window position (bits 7-2 of CR59 or the high order 6 bits of the the PCI Base Address 0). This is concatenated with the display memory address specified by the programmer.

In addition to enabling the new MMIO, the programmer must also enable linear addressing and specify the window size exactly as required for the old linear addressing. Note that since only bits 31-26 are used to specify the base address, A0000H cannot be specified and the 64K banking scheme possible with the old linear addressing cannot be used with the new linear addressing.

When big endian addressing is used, the required byte swapping for linear addressing is specified by bits 2-1 of CR53. This applies to both reads and writes. Alpha pitching is also available for big endian programmers using new linear addressing. This is enabled by setting CR53_0 to 1. Alpha pitching allows the programmer to directly read/write video memory when using a packed 24 bits/pixel mode. The programmer uses standard 32-bit accesses, but the extra (alpha) byte is automatically discarded for writes and added back (as 00H) for reads.

15.3 READ AND WRITE ORDERING

An overview of the ViRGE/VX internal organization is shown in Figure 15-1. Note that there are three independent and concurrent paths for communications between the CPU and ViRGE/VX registers and memory. The time required for any given read or write to complete (latency) varies by path. This can have important implications for the programmer.

First is the issue of write ordering. For example, a linear addressing write to memory uses the command FIFO path, while an image write to memory uses the S3d FIFO path. If the programmer issues a linear addressing command and then an image write command before the linear address command completes (or vise versa), there is no guarantee which will complete first. For total safety from prematurely overwriting memory data, the programmer must check that the S3d FIFO is empty before doing linear addressing updating or for command FIFO empty before doing an image transfer.

Similarly, if correct operation of any command is dependent on operation using another FIFO path (such as a VGA register update before an S3d command), the programmer must ensure that the relevant FIFO is empty before issuing the dependent command.

Reads through the LPB and VGA paths bypass the respective FIFOs. However, they will be held until the relevant FIFO is empty before completing. For PCI systems, this will generate a disconnect (if bit 3 of CR66 is set to 1). This hold guarantees that a read of a register following a write will yield the correct data. Reads of S3d registers go through the S3d FIFO. However, any read with the S3d FIFO not empty or with the S3d Engine busy will yield undefined results.

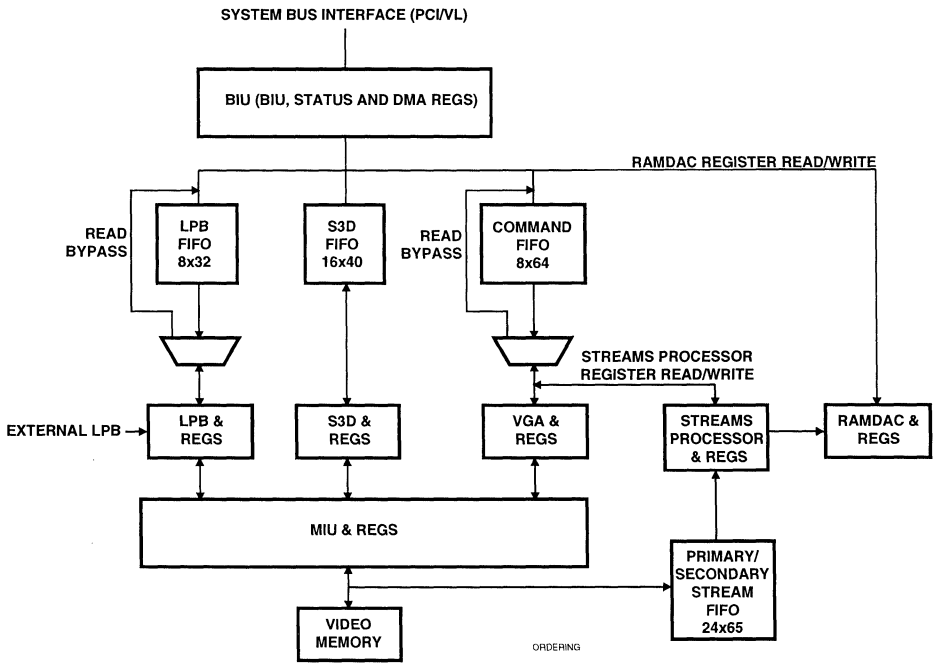


Figure 15-1. Internal Organization

15.4 S3d ENGINE PROGRAMMING

All Enhanced mode programming should be done using memory-mapped I/O.

MMIO Format:

Enable MMIO
Point ES to A000H (old MMIO) or base address (new MMIO)
Load the x and y values into EAX (y value in the low word and x value in the high word), I.e.,

EAX ⇐ x,y

MOV ES:[REGMNEMONIC], EAX

The MMIO scheme is the most efficient and is used where appropriate in the programming examples provided later in this section. All assume that the ES register points to A000H is the old MMIO is being used or the base address if the new MMIO is being used.

15.4.1 Notational Conventions

The following provides examples of the conventions used in the programming examples. Text following a ';' is a comment.

ES:[MMXXXX] ⇐ BN1 (bh-bl), BN2 (bh-bl)

MMXXXX identifies the memory-mapped register, with XXXX being the variable part of the address offset. Thus MMA504 identifies the register at offset 100 A504H. BN1 (bh-bl) represents a bit mnemonic followed by the bit location(s). Thus, SRC_X (26-16) indicates that the Source X value is programmed into bits 26-16.

The complete binary programming of the Command Set register is provided. For example,

ES:[MM????] ⇐ XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX Where:

???? = appropriate variable offset value for the Command Set register, e.g., A500 for BitBLTs.
X = 0; bit value = 0
X = 1; bit value = 1
X = S; this bit value must be specified, but can vary for this command

Image transfers (CPU pixel data writes to the frame buffer) are notated as follows:

COUNT
IMAGEDATA ⇐ RECT_DATA

The COUNT is the number of CPU writes. IMAGEDATA means the 32K Image Data Transfer memory space at the memory-mapped location shown in Table 15-1.

15.4.2 Initial Setup

All examples assume the desired mode is selected. If bit 1 of the Command Set register is set to 1, all bitmap updates are affected by the settings in the clipping registers (MMxxDC, MMxxE0).

15.4.3 Autoexecute

When bit 0 of the Command Set register is cleared to 0, the command is executed when the Command Set register is written. If this bit is set to 1, the command is not executed until the register with the highest address for that command type (BitBLT, Line Draw, etc.) is written. This allows multiple executions of a given command using different parameters without re-programming the Command Set register. Full programming examples for autoexecute on are provided for each command type.

15.4.4 Block Writes

VRAMs provide a block write feature. Each RAM contains a color register, which is programmed before a block write is performed. During the block write, each data bit written to the PD bus is expanded to one byte. A bit of logic 1 is expanded to a byte with the color register value, A bit of logic 0 does not change the corresponding memory byte. Each block write writes 64 bits of data, corresponding to 64 bytes of memory data respectively.

8-column block writes are enabled by setting CR57_7 to 1 and clearing MMAx00_16 to 0. If PD7 is pulled low on reset (0 latched in CR36_7), this indicates no support for 8-column block writes and they cannot be enabled. CR57_5 sets the minimum transfer width for invoking block writes (0 = 16 bytes; 1 = 32 bytes). If a transfer is less than the programmed minimum, block writes will not be used even if they are enabled.

Note that if the memory configuration contains DRAM, software must be careful to not attempt block writes to that DRAM. When the block write function is enabled, block writes are activated automatically based on the type of graphics operation and the mode. Block writes are not supported for 24 bits/pixel modes. The following graphics operations are supported:

Note: all bits listed in the supported operations below are in the 2D Command Set register (MMAx00).

- BitBLT with mono pattern (bits 30_27= 0000b, bits 24-17 = F0H, bit 9 = 1, bit 8 = 1)
- BitBLT with color pattern (bits 30-27 = 0000b, bits 24-27 = F0H
- Transparent text (bits 30_27 = 0000b, bits 24-17 = CCH, bit 9 = 1, bit 7 = 1, bit 6 = 1)
- Rectangle fill with solid color (bits 30_27 = 0010b, bits 24-17 = F0H, bit 9 = 1, bit 8 = 1)
- 2D polygon fill with mono pattern (bits 30_27 = 0101b, bits 24-17 = F0H, bit 9 = 1, bit 8 = 1)

For opaque text, software must do a two pass operation. First, do a rectangle fill of the background, then write transparent text with the foreground color.

The follow considerations must be taken into account:

1. The width of the transfer must be equal to or greater than the number of bytes specified by CR53_5. If it is not, block writes will not be performed. This is automatically detected by the hardware.

2. The hardware will always draw in the positive X direction (from left to right), so MMAx00_25 must be set to 1.

In addition, the destination base address must be 64 bytes aligned (bits 5:0 are '0's). This must be detected by software, after which the block write enable bit can be set. If this is not done, garbage will be written to the screen.

15.4.5 2D Programming Examples

This section provides programming examples for the following Enhanced mode 2D drawing operations:

- BitBLT
- Rectangle Fill
- 2D Line Draw
- 2D Polygon Fill

15.4.5.1 BitBLT

The BitBLT function provides a full implementation of the 256 raster operations as defined by Microsoft for Windows. A listing and explanation of these is provided in Appendix A.

Each raster op has three operands: Source, Pattern and Destination. The Source pixel can be from the screen (current bitmap) or from the CPU (image transfer). When the source is the screen, the pixel depth is always the same for both the source and destination (8, 16, 24 bits/pixel). When the source is the CPU, the pixel can be either color (same source and destination pixel depth) or mono (1 bit/pixel).

The Pattern is an 8x8 array of pixels. A mono pattern is specified in the Mono Pattern 0 and 1 registers. The Pattern Foreground and Background Color registers define the pixel colors. A color pattern is specified in a set of registers starting at offset 100 A100H. The number of registers required depends on the color depth.

The Destination pixel is always the screen (current bitmap) and is always color (multi bits/pixel). This is the pixel that will be overwritten or left unchanged by the result of the operation.

Based on the above definitions, there are 6 valid BitBLT cases:

Color Pattern

- Source = Screen, Color Pixels
- Source = CPU, Color Pixels
- Source = CPU, Mono Pixels

Mono Pattern

- Source = Screen, Color Pixels
- Source = CPU, Color Pixels
- Source = CPU, Mono Pixels

When the source and destination are overlapping rectangles on the screen , care must be taken so that the source data is not overwritten before it is moved. This issue is explained next, followed by programming examples for each of these above cases.

Overlapping Rectangles Case

Figure 15-2 shows the 4 cases for overlapping rectangles. Table 15-2 gives the proper programming parameters for each case. The direction indicates the order in which the pixels are moved, from left to right (X+) or right to left (X-) and top to bottom (Y+) or bottom to top (Y-). These are specified via bits 25 and 26 of the Command Set register. The source and destination coordinates are specified via the Rectangle Source XY and Rectangle Destination XY registers. x1,Y1 is the pixel position of the upper left hand corner of the source rectangle. x2,Y2 is the pixel position of the upper left hand corner of the destination rectangle. The width of the rectangle is W (in pixels) and the height is H (in lines). As indicated in the figure, you always start with the source corner inside the overlap and move that pixel to the corresponding corner for the destination pixel.

Table 15-2 Programming Parameters for Overlapping BitBLTs

Case	Direction	SRC_X	SRC_Y	DEST_X	DEST_Y
1	X+, Y+	x1	y1	x2	y2
2	X-, Y-	x1 + W - 1	y1 + H -1	x2 + W - 1	y2 + H -1
3	X-, Y+	x1 + W - 1	y1	x2 + W - 1	y2
4	X+, Y-	x1	y1 + H -1	x2	y2 + H -1

The basic algorithm is if the drawing direction is negative, add [rectangle dimension -1] in that direction to the normal source/destination location. If the drawing direction is positive, use the original source/destination location.

The parameters for Case 1 are appropriate for non-overlapping rectangles.

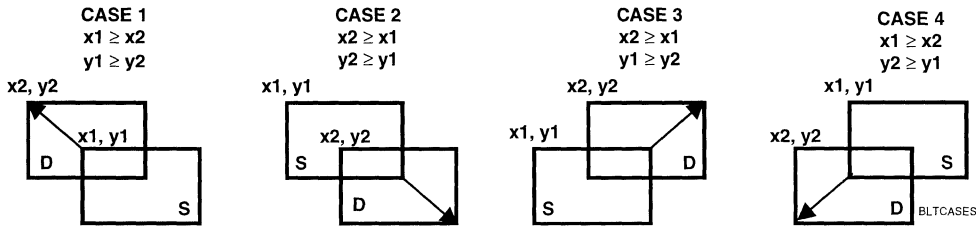


Figure 15-2. Overlapping BitBLT Cases

Color Pattern Case 1 (Source = Screen, Color Pixels)

This command copies a source rectangular area in display memory to another location in display memory. The 8x8 pixel pattern is programmed in the color pattern registers. For this example, assume x1,y1 is the top left corner of the source rectangle in display memory and x2,y2 is the top left corner of the destination rectangle. The rectangles can be overlapping or disjoint. See Table 18-1 for the source and destination coordinate parameter values for overlapping cases. The height and width (in pixels) of the rectangle being copied are H and W. The color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMA100] ⇐ P3 (31-24), P2 (23-16), P1 (15-8), P0 (7-0)          ; Pixels 3-0 of the color pattern
  .
  .
  .
ES:[MMA13C] ⇐ P63 (31-24), P62 (23-16), P61 (15-8), P60 (7-0)      ; pixels 63-60 of the color pattern
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                                ; rectangle width and height
ES:[MMA508] ⇐ SRC_X (26-16), SRC_Y (10-0)                          ; source x and y start coordinates
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)                        ; destination x and y start coord.
ES:[MMA500] ⇐ 0000 0SSS SSSS SSS0 0000 0000 0010 00S0              ; Command Set register
```

The following must be specified: Y direction (bit 26), X direction (bit 25), ROP (bits 24-17), clipping enable (bit 1). Bits 4-2 will be different for other color depths.

Autoexecute On:

```
ES:[MMA100] ⇐ P3 (31-24), P2 (23-16), P1 (15-8), P0 (7-0)          ; pixels 3-0 of the color pattern
  .
  .
  .
ES:[MMA13C] ⇐ P63 (31-24), P62 (23-16), P61 (15-8), P60 (7-0)      ; pixels 63-60 of the color pattern
ES:[MMA500] ⇐ 0000 0SSS SSSS SSS0 0000 0000 0010 00S1              ; bit 0 = 1 for autoexecute
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                                ; rectangle width and height
ES:[MMA508] ⇐ SRC_X (26-16), SRC_Y (10-0)                          ; source x and y start coordinates
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)                        ; destination x and y start coord.
```

The command is executed when MMA50C is programmed. The order of programming the other registers is not important. With autoexecute on, additional BitBLTs can be performed by reprogramming only the parameter registers (not the Command Set register), always ending with the Rectangle Destination XY register (MMA50C).

Color Pattern Case 2 (Source = CPU, Color Pixels)

This command transfers a rectangular color image provided by the CPU to a location in display memory. The 8x8 pixel pattern is programmed in the color pattern registers. For this example, assume the height and width (in pixels) of the rectangle being copied are H and W. The color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMA100] ⇐ P3 (31-24), P2 (23-16), P1 (15-8), P0 (7-0)          ; pixels 3-0 of the color pattern
  .
  .
  .
ES:[MMA13C] ⇐ P63 (31-24), P62 (23-16), P61 (15-8), P60 (7-0)      ; pixels 63-60 of the color pattern
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                                ; rectangle width and height
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)                        ; destination x and y start coord.
ES:[MMA500] ⇐ 0000 000S SSSS SSS0 00SS SS00 1010 00S0              ; Command Set register
```

The following must be specified: ROP (bits 24-17), first dword offset (bits 13-12), image transfer alignment (bits 11-10), clipping enable (bit 1). Bits 4-2 will be different for other color depths.

```
COUNT (of image pixel data to transfer) = (See Note)
IMAGEDATA ⇐ RECT_DATA          ; Output data to Image Transfer addresses for COUNT dwords
```

Autoexecute On:

Writing to the Destination XY register (MMA50C) executes the command.

Note

If the CPU obtains the image data from a source bitmap written to system memory by the application, the application passes the origin of this bitmap, its width, height and color depth. Some or all of this bitmap can then be blitted to display memory (screen). The method of transfer varies depending on whether or not the entire bitmap or a partial bitmap is transferred.

For source bitmaps from an application, each line is required by specification to be word aligned, i.e., data for a new line begins with the next word after the last word containing valid data for the previous line. Therefore, to transfer a complete source bitmap, the driver does the following:

1. All image transfers must be doubleword aligned. Therefore, bits 13-12 of the Command Set register must be programmed to properly reflect the alignment of the first pixel of the source bitmap. For example, if the first pixel of the source bitmap starts with the third byte of the first doubleword-aligned read, bits 13-12 of the Command Set register must be programmed to 10b to tell the Engine to ignore the first two bytes.
2. Word alignment must be specified by programming bits 11-10 of the Command Set register to 01b. This tells the Engine that the data for the next line starts at the next word after the data ending the line. In some cases, doubleword alignment is appropriate (bits 11-10 of the Command set register = 10b). This is more efficient, but is a special case. Word alignment always works.
3. To determine the number of doublewords to transfer, calculate (for the source bitmap):

   int [(width x height x bits/pixel) + 31]/32.

4. The image transfer area in memory is 32K (offset 100 0000H - 100 7FFFH). The driver must monitor the addresses for image writes and reset the address pointer back to the start before any writes are made beyond the 32K area.

If the application requests that only a rectangular subsection of the source bitmap be transferred to display memory, the driver has multiple choices of how to do this.

1. The driver can transfer the entire source bitmap and use the clipping registers to eliminate the unwanted pixels.
2. The driver can transfer only the requested pixels, but it must do this one line at time. If the start of each line is not doubleword aligned, the driver must determine the doubleword address containing the first data for the first line and the number of doublewords required to send the whole line. It must then issue the command to blit this line, with bits 13-12 of the Command Set register set to ignore the appropriate number of bytes at the start of the line. The driver must then change the address to the start of the next line and repeat the above process, including specification of a new destination start address. The result is that one command is executed for each line.

   Note that if the lines for the requested pixels happen to start at doubleword addresses, the entire rectangle can be blitted with a single command because no data needs to be ignored at the start of each line. The driver still needs to keep track of the line length and increment the address by the stride at the end of each line.
3. The driver can transfer the requested pixels as described in 2 above and use the clipping registers to eliminate any extra pixels at the start of each line.

Color Pattern Case 3 (Source = CPU, Mono Pixels)

This command transfers a rectangular mono image provided by the CPU to a location in display memory. The mono image is converted to the screen color depth based on the the pattern color (potentially) mixed with the screen (destination) color. The 8x8 pixel pattern is programmed in the color pattern registers. For this example, assume the height and width (in pixels) of the rectangle being copied are H and W. The screen color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMA100] ⇐ P3 (31-24), P2 (23-16), P1 (15-8), P0 (7-0)          ; pixels 3-0 of the color pattern
  .
  .
  .
ES:[MMA13C] ⇐ P63 (31-24), P62 (23-16), P61 (15-8), P60 (7-0)      ; pixels 63-60 of the color pattern
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                                ; rectangle width and height
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)                        ; destination x and y start coord.
ES:[MMA500] ⇐ 0000 000S SSSS SSS0 00SS SS00 1110 00S0              ; Command Set register
```

The following must be specified: ROP (bits 24-17), first dword offset (bits 13-12), image transfer alignment (bits 11-10), clipping enable (bit 1). Bits 4-2 will be different for other color depths.

```
COUNT (of image pixel data to transfer) = (See Note)
IMAGEDATA ⇐ RECT_DATA              ; Output data to Image Transfer addresses for COUNT dwords
```

Autoexecute On:

Writing to the Destination XY register (MMA50C) executes the command.

Note

If the source bitmap is provided by the application, then the entire Note for the previous color pixels case also applies to this mono pixel case because each line is required to be word aligned. If the source bitmap is provided by the driver, e.g., font data, the driver should byte align the data and program bits 11-10 of the Command Set register to 00b to specify byte alignment to the Engine.

Mono Pattern Case 1 (Source = Screen, Color Pixels)

This command copies a source rectangular area in display memory to another location in display memory. It is identical to the Color Pattern Case 1 except that the 8x8 pixel pattern is programmed in the mono pattern registers and the pattern color is taken from the pattern foreground and background registers. For this example, assume x1,y1 is the top left corner of the source rectangle in display memory and x2,y2 is the top left corner of the destination rectangle. The rectangles can be overlapping or disjoint. See Table 18-1 for the source and destination coordinate parameter values for overlapping cases. The height and width (in pixels) of the rectangle being copied are H and W. The screen color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMACE8] ⇐ MONO PATTERN 0                          ; 1st 32 bits of mono pattern
ES:[MMACEC] ⇐ MONO PATTERN 0                          ; 2nd 32 bits of mono pattern
ES:[MMACF0] ⇐ DATA1 (7-0)                             ; 8-bit pattern backgnd color index
ES:[MMACF4] ⇐ DATA1 (7-0)                             ; 8-bit pattern foregnd color index
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                   ; rectangle width and height
ES:[MMA508] ⇐ SRC_X (26-16), SRC_Y (10-0)             ; source x and y start coordinates
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)           ; destination x and y start coord.
ES:[MMA500] ⇐ 0000 0SSS SSSS SSS0 0000 0001 0010 00S0 ; Command Set register
```

The following must be specified: Y direction (bit 26), X direction (bit 25), ROP (bits 24-17), clipping enable (bit 1). Bits 4-2 and the fields programmed for the background and foreground colors will be different for other color depths.

Autoexecute On:

Writing to the Destination XY register (MMA50C) executes the command.

Mono Pattern Case 2 (Source = CPU, Color Pixels)

This command transfers a rectangular color image provided by the CPU to a location in display memory. It is identical to the Color Pattern Case 1 described earlier except that the 8x8 pixel pattern is programmed in the mono pattern registers and the pattern color is taken from the pattern foreground and background registers. For this example, assume the height and width (in pixels) of the rectangle being copied are H and W. The screen color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMACE8] ⇐ MONO PATTERN 0                          ; 1st 32 bits of mono pattern
ES:[MMACEC] ⇐ MONO PATTERN 0                          ; 2nd 32 bits of mono pattern
ES:[MMACF0] ⇐ DATA1 (7-0)                             ; 8-bit pattern backgnd color index
ES:[MMACF4] ⇐ DATA1 (7-0)                             ; 8-bit pattern foregnd color index
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                   ; rectangle width and height
ES:[MMA508] ⇐ SRC_X (26-16), SRC_Y (10-0)             ; source x and y start coordinates
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)           ; destination x and y start coord.
ES:[MMA500] ⇐ 0000 000S SSSS SSS0 0001 1001 0010 00S0 ; Command Set register
```

The following must be specified: ROP (bits 24-17), first dword offset (bits 13-12), image transfer alignment (bits 11-10), clipping enable (bit 1). Bits 4-2 will be different for other color depths.

Autoexecute On:

```
COUNT (of image pixel data to transfer) = (See Note)
IMAGEDATA ⇐ RECT_DATA          ; Output data to Image Transfer addresses for COUNT dwords
```

Autoexecute On:

Writing to the Destination XY register (MMA50C) executes the command.

Note

If the CPU obtains the image data from a source bitmap written to system memory by the application, the application passes the origin of this bitmap, its width, height and color depth. Some or all of this bitmap can then be blitted to display memory (screen). The method of transfer varies depending on whether or not the entire bitmap or a partial bitmap is transferred.

For source bitmaps from an application, each line is required by specification to be word aligned, i.e., data for a new line begins with the next word after the last word containing valid data for the previous line. Therefore, to transfer a complete source bitmap, the driver does the following:

1. All image transfers must be doubleword aligned. Therefore, bits 13-12 of the Command Set register must be programmed to properly reflect the alignment of the first pixel of the source bitmap. For example, if the first pixel of the source bitmap starts with the third byte of the first doubleword-aligned read, bits 13-12 of the Command Set register must be programmed to 10b to tell the Engine to ignore the first two bytes.
2. Word alignment must be specified by programming bits 11-10 of the Command Set register to 01b. This tells the Engine that the data for the next line starts at the next word after the data ending the line. In some cases, doubleword alignment is appropriate (bits 11-10 of the Command set register = 10b). This is more efficient, but is a special case. Word alignment always works.

3. To determine the number of doublewords to transfer, calculate (for the source bitmap):

   int [(width x height x bits/pixel) + 31]/32.

4. The image transfer area in memory is 32K (offset 100 0000H - 100 7FFFH). The driver must monitor the addresses for image writes and reset the address pointer back to the start before any writes are made beyond the 32K area.

If the application requests that only a rectangular subsection of the source bitmap be transferred to display memory, the driver has multiple choices of how to do this.

1. The driver can transfer the entire source bitmap and use the clipping registers to eliminate the unwanted pixels.
2. The driver can transfer only the requested pixels, but it must do this one line at time. If the start of each line is not doubleword aligned, the driver must determine the doubleword address containing the first data for the first line and the number of doublewords required to send the whole line. It must then issue the command to blit this line, with bits 13-12 of the Command Set register set to ignore the appropriate number of bytes at the start of the line. The driver must then change the address to the start of the next line and repeat the above process, including specification of a new destination start address. The result is that one command is executed for each line.

   Note that if the lines for the requested pixels happen to start at doubleword addresses, the entire rectangle can be blitted with a single command because no data needs to be ignored at the start of each line. The driver still needs to keep track of the line length and increment the address by the stride at the end of each line.
3. The driver can transfer the requested pixels as described in 2 above and use the clipping registers to eliminate any extra pixels at the start of each line.

Mono Pattern Case 3 (Source = CPU, Mono Pixels)

This command transfers a rectangular mono image provided by the CPU to a location in display memory. The mono image is converted to the screen color depth based on the the pattern color (potentially) mixed with the screen (destination) color. It is identical to the Color Pattern Case 3 described earlier except that the 8x8 pixel pattern is programmed in the mono pattern registers and the pattern color is taken from the pattern foreground and background registers. For this example, assume the height and width (in pixels) of the rectangle being copied are H and W. The screen color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMACE8] ⇐ MONO PATTERN 0                          ; 1st 32 bits of mono pattern
ES:[MMACEC] ⇐ MONO PATTERN 0                          ; 2nd 32 bits of mono pattern
ES:[MMACF0] ⇐ DATA1 (7-0)                             ; 8-bit pattern backgnd color index
ES:[MMACF4] ⇐ DATA1 (7-0)                             ; 8-bit pattern foregnd color index
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                   ; rectangle width and height
ES:[MMA508] ⇐ SRC_X (26-16), SRC_Y (10-0)             ; source x and y start coordinates
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)           ; destination x and y start coord.
ES:[MMA500] ⇐ 0000 000S SSSS SSS0 0000 0001 1110 00S0 ; Command Set register
```

The following must be specified: ROP (bits 24-17), first dword offset (bits 13-12), image transfer alignment (bits 11-10), clipping enable (bit 1). Bits 4-2 will be different for other color depths.

```
COUNT (of image pixel data to transfer) = (See Note)
IMAGEDATA ⇐ RECT_DATA          ; Output data to Image Transfer addresses for COUNT dwords
```

Autoexecute On:

Writing to the Destination XY register (MMA50C) executes the command.

Note

If the source bitmap is provided by the application, then the entire Note for the previous color pixels case also applies to this mono pixel case because each line is required to be word aligned. If the source bitmap is provided by the driver, e.g., font data, the driver should byte align the data and program bits 11-10 of the Command Set register to 00b to specify byte alignment to the Engine.

15.4.5.2 Rectangle Fill

This command draws a filled rectangle on the screen. Only ROPs that do not contain a source can be used. If the ROP contains a pattern, the pattern specification will be ignored. Instead, the pattern value is forced to a 1 by the hardware, selecting the pattern foreground color. ROPs specifying only the destination (screen) and optionally a logical operation (e.g., NOT D) can be used. In this case, the rectangle color will depend only on the current screen color. For this example, assume the height and width (in pixels) of the rectangle being drawn are H and W. The screen color depth is assumed to be 8 bits/pixel.

Autoexecute Off:

```
ES:[MMA4F4] ⇐ DATA1 (7-0)                                  ; 8-bit pattern foregnd color index
ES:[MMA504] ⇐ W-1 (26-16), H (10-0)                        ; rectangle width and height
ES:[MMA50C] ⇐ DEST_X (26-16), DEST_Y (10-0)                ; destination x and y start coord.
ES:[MMA500] ⇐ 0001 000S SSSS SSS0 0000 0001 0010 00S0      ; Command Set register
```

The following must be specified: ROP (bits 24-17), clipping enable (bit 1). Bits 4-2 will be different for other color depths. Bit 8 must be set to 1 to specify a mono pattern.

Autoexecute On:

Writing to the Destination XY register (MMA50C) executes the command.

15.4.5.3 2D Line Draw

This command draws a two-dimensional line on the screen. Only ROPs that do not contain a source can be used. If the ROP contains a pattern, the pattern specification will be ignored. Instead, the pattern value is forced to a 1 by the hardware, selecting the pattern foreground color. ROPs specifying only the destination (screen) and optionally a logical operation (e.g., NOT D) can be used. In this case, the line color will depend only on the current screen color. Assume x1,y1 are the starting coordinates of the requested line and x2,y2 are the ending coordinates. x1 and x2 are pixel coordinates, with 0 being the x coordinate of the first (leftmost) pixel on each line. y1 and y2 are line coordinates, with 0 being the coordinate of the first (topmost) line.

The S3d Engine draws 2D lines from the bottom up, regardless of the requested drawing direction. Figure 15-3 shows four cases of requested lines (shown by the arrows on the grids). In Case 1, the requested drawing direction is the same as is used by the S3d Engine, so the x1,y1 coordinates are used to determine the starting coordinates (X_{START}, Y_{START}). In Case 2, the line will be drawn by the S3d Engine exactly reversed from that requested, so x2,y2 are used to determine the starting coordinates. In these and the other two cases, the small arrows outside the grid point to the starting coordinates used by the S3d Engine. The programmer must always use the end with the largest y value as the starting point.

Another complexity is illustrated by Case 1. If the line is X MAJOR (i.e., for a given movement along the line, the x value increases faster than the y value), the starting x value must be adjusted to the point indicated by the intersection of the dashed lines. This is a 1/2 pixel (x direction) extension from the first pixel to be drawn. For Y MAJOR lines (Case 4), this adjustment is not required.

The parameters required to draw a line must be calculated by software and programmed into the appropriate registers. The first values that must be calculated are:

ΔX = x2 - x1 or x1 - x2
ΔY = y2 - y1 or y1 - y2

The important point is that if x2 - x1 is used for ΔX, then y2 - y1 must be used for ΔY and vice versa.

The parameters required are:

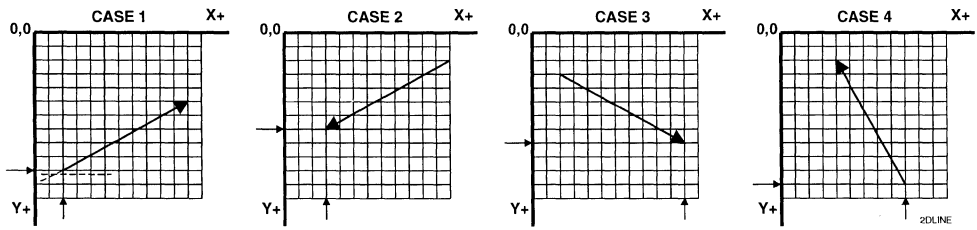


Figure 15-3. 2D Line Drawing Cases

X DELTA = - (ΔX << 20)/ΔY (integer divide)

This is value is programmed in MMA970 with bit 31 as the sign bit (0 = positive)

X START = (x_{START} << 20) + (X DELTA/2) for X MAJOR lines and positive X DELTA
X START = (x_{START} << 20) + (X DELTA/2) + ((1 << 20) - 1) for X MAJOR lines and negative X DELTA
X START = (x_{START} << 20) for Y MAJOR lines

This value is programmed in MMA974 with bits 31 and 30 as sign bits. The preceding discussion describes how to determine x_{START}.

Y START = y_{START}

This value is programmed in MMA978_10-0. It is the y value of the first scan line and is always the largest requested y.

Y COUNT = [abs (y2 -y1)] + 1

This value is programmed in MMA97C_10-0. It is the number of scanlines to draw.

The horizontal drawing direction is specified in MMA97C_31 (0 = right to left; 1 = left to right)

The final parameters to be specified are used primarily for the case where the programmer is drawing a polyline (connected line segments) and specifies "last pixel not drawn" for one segment. This is done so that the last pixel of one segment is not drawn a second time as the first pixel of the next segment. The parameters are:

END1 = x coordinate for the last pixel to be drawn for the line (MMA96C_15-0)
END0 = x coordinate for the first pixel to be drawn for the line (MMA96C_31-0)

The both cases, the 5 most significant bits are sign bits and must be 0's to indicate a positive value.

The complication here is again that the S3d Engine drawing direction may not be the same as the requested direction. In Case 1 of Figure 15-3, the two directions are the same. If "last pixel off" is specified, then END0 is programmed with the x1 (requested starting x) value and END1 with x2 -1 (one

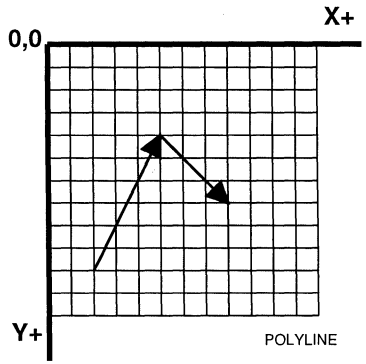


Figure 15-4. Polyline Drawing Example

less than the requested ending x value to stop the line one pixel short). In Case 2, the directions are opposite. END0 is programmed with x2 +1 and END1 with x1. Thus, the S3d Engine (which starts at the requested ending x position so it can draw upward) skips the first pixel and draws the last to accommodate the reversed drawing direction. In a similar fashion, is is easy to see that for Case 3, END0 is x2 - 1 and END1 is x1. For Case 4, END0 is x1 and END1 is x2 +1.

If "last pixel off" is not requested, the END0 and END1 values are the same as described above except that 1 is not added or subtracted as appropriate. Thus, the full x values of both ends of the line are specified. This allows a horizontal line to be drawn. Normally, the X DELTA value for a horizontal line would be infinity ($\Delta Y = 0$). For this case, the programmer can specify an X DELTA of 0 and the S3d engine will use the endpoint parameters to draw the correct line.

The following programming example is for a polyline as shown in Figure 15-4. The first requested segment goes up to the right with the last pixel not drawn. The second segment goes down to the right with all pixels drawn. This first segment must be drawn first since it has the largest y value. It is drawn as described for Case 1 in Figure 15-3 except the line is X MAJOR. The second line segment is drawn as described for Case 3. This line is neither X MAJOR or Y MAJOR, so the Y MAJOR assumption should be used because it is simpler to calculate X START. Autoexecute is used so that the Command Set register does not need to be re-programmed.

```
                                                    ; 1st line segment
ES:[MMA96C] ⇐ END0 (31-16), END1 (15-0)             ; last pixel off for END1
ES:[MMA970] ⇐ X DELTA                               ; x direction gradient
ES:[MMA974] ⇐ X START                               ; starting x coord. for S3d Engine
ES:[MMA978] ⇐ Y START (10-0)                        ; starting y coord. for S3d Engine
ES:[MMA900] ⇐ 0001 100S SSSS SSS0 0000 0000 0010 00S1  ; Command Set (autoexecute)
ES:[MMA97C] ⇐ DIR (31), Y COUNT (10-0)              ; draw dir and # of scanlines
                                                    ; 2nd line segment
ES:[MMA96C] ⇐ END0 (31-16), END1 (15-0)             ; all pixels drawn
ES:[MMA970] ⇐ X DELTA                               ; x direction gradient
ES:[MMA974] ⇐ X START                               ; starting x coord. for S3d Engine
ES:[MMA978] ⇐ Y START (10-0)                        ; starting y coord. for S3d Engine
ES:[MMA97C] ⇐ DIR (31), Y COUNT (10-0)              ; draw dir and # of scanlines
```

Note that with autoexecute on (bit 0 of the Command Set register set to 1), a line is drawn every time MMA97C is programmed. Also note that the Command Set register has a unique address for each command type, e.g., it is at offset A900 for 2D lines while it is at A500 for BitBLTs and rectangle fills. Only the ROP (bits 24-17) and clipping (bit 1) are optionally specified for line draws.

To draw a disconnected line after drawing a polyline, autoexecute must first be turned off. This is done by writing to the Command Set register with bit 0 cleared to 0 and the command (bits 30-27) specified as 1111b (NOP).

15.4.5.4 2D Polygon Fill

This command is used to generate a filled polygon. Any number of edges can be drawn, but the shape must be such that any horizontal line must intersect the polygon edges in no more than two places. The exception is that any edge can be horizontal. Only ROPs that do not contain a source can be used. If the ROP contains a pattern, the pattern specification will be taken from the appropriate color or mono pattern registers. ROPs specifying only the destination (screen) and optionally a logical operation (e.g., NOT D) can be used. In this case, the pixel color will depend only on the current screen color for the destination pixel.

For polygon fills, the end points of each edge segment are not explicitly specified and cannot be optionally drawn or not drawn. Drawing of the overlapping pixels is handled automatically. Also, instead of specifying the direction of line drawing, the edge or edges to be updated are specified via bits 28 and 29 of MMAD7C. Otherwise, the parameters for each line are calculated exactly as for 2D lines.

$\Delta X = x2 - x1$ or $x1 - x2$
$\Delta Y = y2 - y1$ or $y1 - y2$

The important point is that if x2 - x1 is used for ΔX, then y2 - y1 must be used for ΔY and vice versa.

The parameters required are:

X DELTA = - (ΔX << 20)/ΔY (integer divide) - right and left edges

These values are programmed in MMAD68 and MMAD70 with bit 31 as the sign bit (0 = positive)

X START = (x_{START} << 20) + (X DELTA/2) + (1 << 19) for X MAJOR lines - right and left edges
X START = (x_{START} << 20) + (1 << 19) for Y MAJOR lines - right and left edges

These values are programmed in MMAD6C and MMAD74 with bits 31 and 30 as sign bits. The line draw discussion describes how to determine x_{START}.

Y START = y_{START}

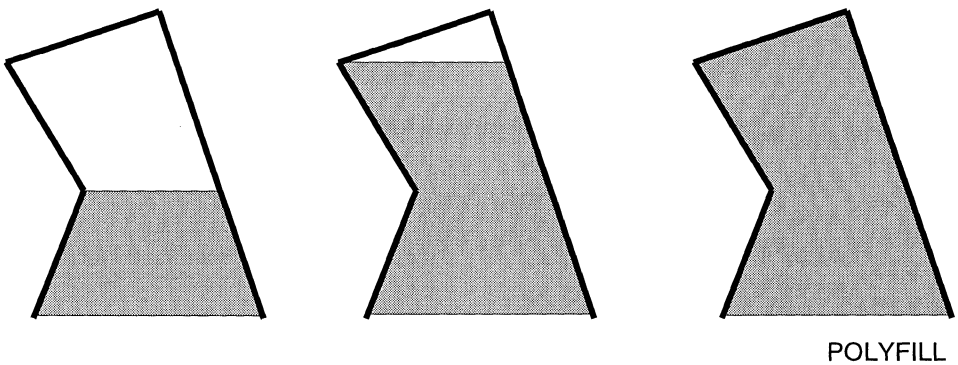


Figure 15-5. Polygon Fill Example

This value is programmed in MMA978_10-0. It is the y value of the first scan line and is always the largest requested y.

Y COUNT = [abs (y2 -y1)] + 1

This value is programmed in MMAD78_10-0. It is the number of scanlines to draw for each edge segment.

The S3d Engine draws polygons from the bottom up as shown in the example in Figure 15-5. In the first iteration, the programmer specifies line parameters for the left and right edges and specifies that they both be updated. The first iteration also specifies the number of scan lines up to the first vertex, which is on the left edge in this example. This results in the trapezoid shown in the leftmost example. The second iteration only specifies the second segment of the left edge, resulting in the middle example. Since the right edge does not change slope, it should not be re-specified or updated (MMAD7C_28 = 0). This speeds the drawing by eliminating the need for a recalculation for that edge. The third iteration draws the third segment of the left edge, which joins the right edge to complete the polygon as shown by the right hand example. Again, the right edge should not be re-specified or updated.

As with the bottom edge shown in the example, if the top edge is a horizontal line, that line does not have to be drawn to close the polygon.

```
                                                    ; 1st iteration
ES:[MMAD68] ⇐ RIGHT EDGE X DELTA                    ; right edge x direction gradient
ES:[MMAD6C] ⇐ RIGHT EDGE X START                    ; right edge starting x coord.
ES:[MMAD70] ⇐ LEFT EDGE X DELTA                     ; left edge x direction gradient
ES:[MMAD74] ⇐ LEFT EDGE X START                     ; left edge starting x coord.
ES:[MMAD78] ⇐ Y START (10-0)                        ; bottommost y value
ES:[MMAD00] ⇐ 0010 100S SSSS SSS0 0000 0000 0010 00S1   ; Command Set (autoexecute)
ES:[MMAD7C] ⇐ Update Lft (29), Update Rgt (28), Y COUNT (10-0) ; update edge and # of scanlines
                                                    ; 2nd iteration
ES:[MMAD70] ⇐ LEFT EDGE X DELTA                     ; left edge x direction gradient
ES:[MMAD74] ⇐ LEFT EDGE X START                     ; left edge starting x coord.
ES:[MMAD7C] ⇐ Update Lft (29), Update Rgt (28), Y COUNT (10-0) ; update edge(s) and # of scanli-
nes
                                                    ; 3rd iteration
ES:[MMAD70] ⇐ LEFT EDGE X DELTA                     ; left edge x direction gradient
ES:[MMAD74] ⇐ LEFT EDGE X START                     ; left edge starting x coord.
ES:[MMAD7C] ⇐ Update Lft (29), Update Rgt (28), Y COUNT (10-0) ; update edge and # of scanlines
```

Note that with autoexecute on (bit 0 of the Command Set register set to 1), a trapezoid fill is executed every time MMAD7C is programmed. Also note that the Command Set register has a unique address for each command type, e.g., it is at offset AD00 for 2D polygon fills while it is at A500 for BitBLTs and rectangle fills and A900 for 2D lines. Only the ROP (bits 24-17) and clipping (bit 1) are optionally specified for polygon fills.

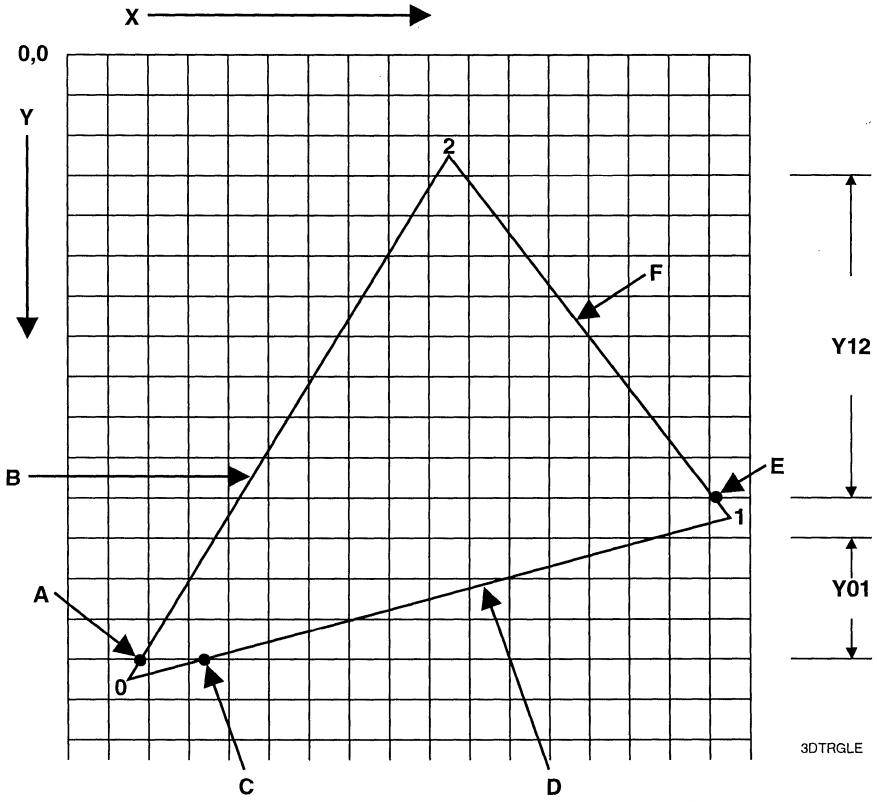


Figure 15-6. 3D Triangle Example

15.4.6 3D Graphics Drawing

The S3d Engine accelerates the drawing of 3D lines and triangles. Texturing of 3D triangles and fogging and alpha blending of both 3D lines and 3D triangles is also supported. This section describes the basic 3D drawing capabilities and the register values required to generate the desired image. Programming code is quite complex for 3D operations and will be provided by S3 to customers desiring to create custom drivers.

15.4.6.1 3D Line Drawing

3D line drawing is very similar to 2D line drawing except:

- There is a third (Z) dimension, with increasing values going away from the viewer (into the screen). Like the X value, this is specified in fractional coordinates. (The Y value is always an

integer number of scan lines.) The registers associated with this dimension are 3dZ and 3ZStart and are used only when Z-buffering is desired.

- There are 4 color coordinates for the start of the line and associated color deltas. The color values are Alpha (transparency/opacity factor), Red, Green and Blue. These are all expressed as fractional values. The registers associated with these colors are 3dGdY_dBdY and 3dAdY_dRdY (deltas) and 3GS_BS and 3AS_RS (starts).

15.4.6.2 3D Triangle Drawing

Figure 15-6 represents a typical triangle drawn into the frame buffer. The grid represents pixel coordinates, i.e., each intersection is the location of one pixel. The origin of the grid is at the top left (0,0), with the X dimension increasing to the right and the Y dimension increasing downward. The specified triangle does not have to start or end on a pixel coordinate, as illustrated in the figure.

Vertices 0 through 2 of the triangle to be drawn are numbered by decreasing Y value, i.e., from bottom to top. The triangle is always rendered from bottom to top, starting at the first scan line at or above the starting (bottom) vertex and ending at the last scan line at or below the ending (top) vertex. The location of the 02 side (largest Y dimension) determines the horizontal rendering direction. For a triangle as shown in Figure 15.6, with the 02 side on the left, rendering must be done from left to right. This is specified by setting bit 31 of MMB517C to 1. If the triangle in Figure 15.6 is flipped horizontally so the 02 side is on the right, the rendering direction must be specified as from right to left. This is done by clearing bit 31 of MMB51C to 0.

As many as 43 registers may be required to completely specify the rendering of one 3D triangle with texturing applied. These registers are described in Section 19. Figure 15.6 helps to explain the relevance of most of these registers.

The following registers are associated with point A.

Spatial Dimensions (Point A)	Color Dimensions (Point A)	Texture Dimensions (Point A)
TXStart02 TYStart TZS02	TGS_BS TAS_RS	TDS TUS TVS TWS

The following registers are associated with the Y axis and side 02. Note that the Y component of side 02 (B in Figure 15-6), always determines the number of scan lines required to render the triangle.

Spatial Dimensions (Y axis)	Color Dimensions (Y axis)	Texture Dimensions (Y axis)
TdXdY02 TdZdY TY01_Y12	TdGdY_dBdY TdAdY_dRdY	TdDdY TdUdY TdVdY TdWdY

The TXEnd01 register is associated with point C in Figure 15.6.

The following registers are associated with the X axis and side 01. Note that the X component of side 01 (D in Figure 15-6), is always the maximum width of the rendered triangle.

Spatial Dimensions (X axis)	Color Dimensions (X axis)	Texture Dimensions (X axis)
TdXdY01 TdZdX	TdGdX_dBdX TdAdX_dRdX	TdDdX TdUdX TdVdX TdWdX

The TXEnd12 register is associated with point E in Figure 15.6.

The TdXdY12 register is associated with side 12 (F in Figure 15.6).

The TbU and TbV registers contain the common offset values for the U and V texture dimensions, i.e., these values are added to all U and V specifications.

Triangles can be drawn with perspective correction (bits 30-27 of the Command Set register = 0101 or 0110). Perspective correction uses the W parameters. In addition, the U and V parameters have different bit codings when perspective correction is specified than when it is not. These are explained in the register descriptions. Using automatic perspective correction will normally cause some decrease in performance, but can in some circumstances provide dramatic increases in picture quality.

15.4.7 Z-Buffering

Z-buffering allows the programmer to eliminate rendering of hidden lines and surfaces. It is enabled when bits 25-24 of the Command Set register are 00b and bits 22-20 of the Command Set register are not 000b. Use of z-buffering requires that space be allocated in video memory for the z-buffer. The starting location is specified in the Z_BASE register. For each graphics pixel, the z-buffer contains a corresponding 16 bits of depth information. Bits 22-20 of the Command Set register specify the relational operator used to compare the z value of the source pixel with its corresponding z-buffer value, as follow:

000 = Z compare never passes
001 = Pass if Zs > Zzb
010 = Pass if Zs = Zzb
011 = Pass if Zs ≥ Zzb
100 = Pass if Zs < Zzb
101 = Pass if Zs ≠ Zzb
110 = Pass if Zs ≤ Zzb
111 = Z compare always passes

For example, a setting of 110 means that the source pixel will replace the current pixel in video memory only if its source z value is less than the corresponding z-buffer value. This is the normal comparison, as it allows the pixel closer to the viewer to be drawn. If bit 23 of the Command Set register is set to 1, the source pixel z value will replace the current z-buffer value. If bit 23 of the Command Set register is cleared to 0, the z-buffer value remains unchanged.

The z-buffer comparison occurs before any of the pixel coloring operations described below. If the z comparison fails, no further coloring operations will be done on that pixel. Similarly, if the operator is set to never pass, z-buffering is effectively disabled. This can improve performance.

15.4.8 MUX Buffering

Z-buffering requires 16 bits of video memory storage for each displayable pixel. If insufficient memory is available, MUX buffering may allow z-buffering to be performed. With MUX buffering, the active frame buffer area (draw buffer) is alternately programmed with z-buffer values and pixel colors. This requires that all the primitives (lines and triangles) of the scene be rendered twice, which decreases performance. Otherwise, MUX buffering produces the effects as normal z-buffering.

MUX buffering can only be used when the destination format is 16 bits/pixel and no alpha blending is to be performed (bit 19 of the Command Set register = 0). When the destination format is 16 bits/pixel, bit 15 = 1 indicates the word contains a z value and bit 15 = 0 indicates the word contains an RGB555 value.

With MUX buffering, double buffering should be used so that the z-buffering can be done in the inactive (back) buffer. See the Streams Processor section for an explanation of double buffering. Z-buffering is enabled as explained in the previous section except that bit 23 of the Command Set register must be set to 1 so that the source pixel z value will replace the current z-buffer value. As a final setup step, the entire buffer must be written with either a solid color or a prerendered bitmap. This sets the z bit of each word to 0, indicating that colors are stored.

On the first pass, bits 25-24 of the Command Set register are programmed to 01b to specify the z-buffer pass. The S3d Engine interpolates only the z values of the the source primitive (line or triangle). For each source pixel, if the corresponding destination pixel is a color (bit 15 = 0), the source z value replaces the destination color. For the first primitive to be drawn for the scene, the source pixels (z values) will replace all the corresponding destination pixels (colors) because of the initialization to colors. For subsequent primitives for the scene, the source pixel may or may not replace the destination pixel. It will always replace it if the destination is a color, but if the destination is a z-value, it will only replace

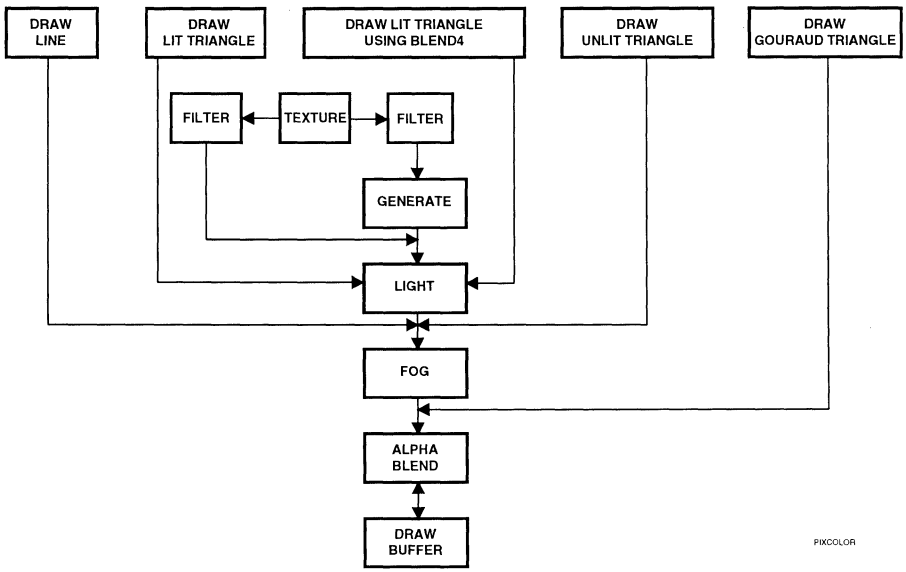


Figure 15-7. Pixel Coloring

it if the z comparison passes. At the end of this pass, all pixels corresponding to primitives are set to z values. All other pixels retain the initialization color values.

For the second pass, bits 25-24 of the Command Set register are programmed to 10b to specify the draw buffer pass. The S3d Engine again interpolates the z values for all source primitives. If the destination pixel is a color, that pixel color is left unchanged. If the destination pixel is a z value, the source z value is compared with the destination z value. If they are equal, the source color is computed and that color value replaces the destination z value. At the end of this pass, all pixels in the buffer contain color values. The buffer is then switched to the front (active) and is used for the next screen refresh.

15.4.9 3D Pixel Color Generation

Pixel color generation for 3D drawing occurs in a series of steps as depicted in Figure 15-7. The first of these, calculate the source pixel color, has been explained in the 3D line and triangle drawing sections above. The remaining steps are:

1. Filter - If texturing is enabled for a 3D triangle, two, four or eight texels (texture pixel) from the texture map can be filtered (interpolated) to generate a texture color to be mixed with the source color in step 3 or a code to be used in the next step.
2. Generate - For certain applications, textures can be stored in a compact colorless mode (Blend4). This step generates a texture color based on the compact coding, which may or may not be the output of filtering from the previous step. This color is used in the next step.
3. Light - If a lit texture triangle is specified, the source pixel color is mixed with the texel color to generate a color which can optionally be fogged or alpha blended.
4. Fog - Also called depth cueing. As shown in Figure 15-7, the input can either be the source pixel color or the result of the filter/generate steps.

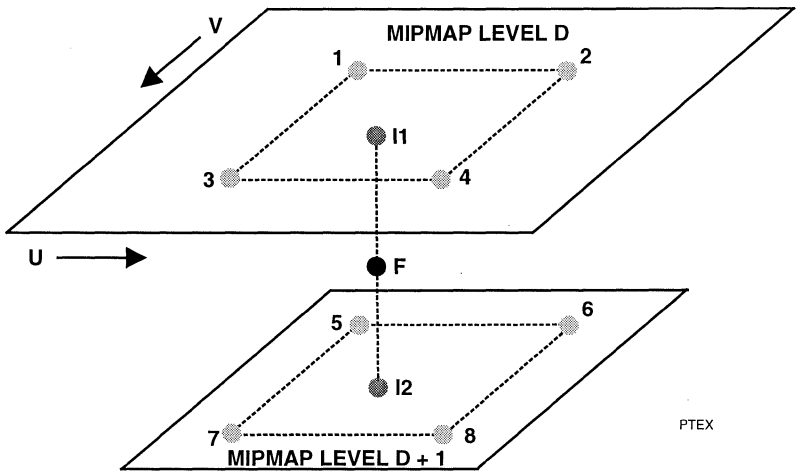


Figure 15-8. Texture Filtering

5. Alpha Blend - The source pixel color or the output of the fogging step (which may be disabled) is blended with the destination pixel color in video memory. This can produce a transparency effect.

Each of these steps is explained in more detail in the following sections.

15.4.9.1 Texture Filtering

Textures are stored in off-screen video memory at a location specified in MMB4EC. The integer components of the U and V parameters generate the memory addresses for each texture element, which is called a texel. The fractional part of the U and V parameters are used in the filter stage for interpolation between texel colors. The texture color format is specified in bits 7-5 of the Command Set register and can be one of the following:

000 = 32 bits/pixel (ARGB8888)
001 = 16 bits/pixel (ARGB4444)
010 = 16 bits/pixel (ARGB1555)
011 = 8 bits/pixel (Alpha4, Blend4)
100 = 4 bits/pixel (Blend4, low nibble)
101 = 4 bits/pixel (Blend4, high nibble)
110 = 8 bits/pixel (palettized)
111 = YU/YV (16 bits/pixel equivalent)

The texture can be a single rectangular pattern or a mipmap. A mipmap contains multiple versions of the same texture, each at successively lower resolutions (1/2, 1/4, 1/8, etc.). The size of the largest mipmap level (level 0) must be specified via bits 11-8 of the Command Set register. The integer part of the D parameter points to the mipmap level to be used for the texture. The fractional part of the D parameter is used for filtering of colors between mipmap levels.

A variety of filter modes are provided via bits 14-12 of the Command Set register, as follows:

000 = M1TPP (MIP_NEAREST)
001 = M2TPP (LINEAR_MIP_NEAREST)
010 = M4TPP (MIP_LINEAR)
011 = M8TPP (LINEAR_MIP_LINEAR)
100 = 1TPP (NEAREST)
101 = V2TPP (used for YU/YV video format)
110 = 4TPP (LINEAR)
110 = Reserved

Modes starting with M are mipmapped. Those without have a single texture level. XTPP means X texels are interpolated per source pixel. Figure 15-8 demonstrates the effect of the 011 setting (M8TPP). The U,V and D parameters point to the texture map location indicated by the black dot at F. To generate the color for this location, the four nearest pixels in mipmap level D (1 - 4) are interpolated to generate the color indicated by the top medium gray dot (I1). The four nearest pixels in mipmap level D + 1 (5 - 8) are interpolated to generate the color indicated by the bottom medium gray dot (I2). The colors at I1 and I2 are then interpolated to produce the final color at F.

If M1TPP or 1TPP is selected, the texel nearest to the programmed texture location is chosen to provide the texture color. For M2TPP, the color is interpolated between the nearest texels from 2 mipmap levels (e.g., texels 1 and 5 in Figure 15-8). For M4TPP or 4TPP, texels 1, 2 , 3 and 4 are interpolated. For V2TPP, which is used only for YUV data, texels 1 and 3 are interpolated.

Filtering of 8 bits/pixel palettized data produces uncertain results. Palettized texel colors can be used if the filter mode is M1TPP or 1TPP (only one texel is used to generate the color) and the texture blending mode (lighting) is specified as decal. This means the texel color replaces the source pixel color (no mixing). Because the color is now palettized, it cannot be texture lit, fogged or alpha blended.

15.4.9.2 Generation

ViRGE/VX provides several compact texture storage modes, called Blend4 (high and low nibble) and Alpha4/Blend4. Blend4 uses 4 bits to define the color for each texel. These bits can be in either the high or low nibble of each byte, allowing the programmer to locate texels from two different textures in a single byte. Alpha4/Blend4 has 4 bits of Alpha coding and 4 bits of RGB color coding in each byte.

Blend4 is useful for textures with a narrow range of colors, such as grass. The 4-bit value is an interpolation factor between two RGB colors defined in the Color 0 (MMB4F8) and Color1 (MMB4FC) registers.

Alpha4/Blend4 is useful for textures with a limited range of colors and transparency, such as a cloudy sky. In this case, there are a few shades of blue-white, with whiter clouds being more opaque than bluer sky. Alpha blending is explained below.

Generation of colors for Blend4 modes occurs after the filter phase. Therefore it is possible to filter multiple Blend4 texels to produce a composite color interpolation factor to be used in the generate phase. The results of this might be hard to predict. The filter phase can be bypassed by selecting a 1TPP filter mode.

15.4.9.3 Lighting

Lighting is the blending of the texel color with the source pixel color. As seen in Figure 15-7, it is used only when a lit triangle is specified in bits 30-27 of the Command Set register. Bits 16-15 of the Command Set register specify the blending modes as follows:

00 = Complex reflection
01 = Modulate
10 = Decal
11 = Reserved

Complex reflection adds the (normalized, 0 = black and 1 = white) texel and pixel colors, with a maximum value of 1. This lightens the pixel.

Modulate multiplies the normalized color values. This results in a smaller value (darker pixel). The programmer may need to compensate for this darkening effect.

Decal replaces the source pixel color with the texel color, essentially overlaying the texture on the scene. This is the only mode that can be used with palettized data.

If the texture map is smaller than the area to be textured, texture wrapping can be turned on via bit 26 of the Command Set register. This allows the texture to be tiled across the scene. If texture wrapping is disabled and and the texture map is smaller than the area to be textured, the texel color is taken from the Texture Border Color register (MMB4F0) for all pixels beyond the texture.

15.4.9.4 Fogging

Fogging is enabled via bit 17 of the Command Set register. This operation uses the pixel's alpha value to interpolate between the pixel color at this stage of the coloring process (see Figure 15-7) and a fog color specified in MMB(0/4)F4. If the alpha value corresponds to the distance from the viewer, this is called depth cueing. If fogging is being done, source alpha cannot be specified for alpha blending (i.e., bits 19-18 of the Command set register cannot be 11b).

15.4.9.5 Alpha Blending

Alpha blending blends the pixel color at this stage of coloring (see Figure 15-7) with the color of the corresponding pixel in the draw buffer. It is enabled via bits 19-18 of the Command Set register. If these bits are 10b, the texture alpha is used for the interpolation factor. The texture alpha is actually the alpha for the pixel at this stage of the coloring and not a texel alpha. If bits 19-18 are 11b, the source alpha is used for the interpolation factor. This is the original pixel alpha before texturing.

Alpha blending is used for transparency effects. The smaller the value of alpha, the more the destination color will dominate the final color (or higher transparency). To be effective, primitives must be drawn in order of increasing transparency, i.e., decreasing alpha.

15.5 PROGRAMMABLE HARDWARE CURSOR

A programmable cursor is supported which is compatible with the Microsoft Windows (bit 4 of CR55 = 0) and X11 (bit 4 of CR55 = 1) cursor definitions. The cursor size is 64 pixels wide by 64 pixels high, with the cursor pattern stored in an off-screen area of display memory. Two monochrome images 64 bits wide by 64 bits high (512 bytes per image) define the cursor shape. The first bit image is an AND mask and the second bit image is an XOR mask. The following is the truth table for the cursor display logic.

AND Bit	XOR Bit	Displayed (Microsoft Windows)	Displayed (X11)
0	0	Cursor Background Color	Current Screen Pixel
0	1	Cursor Foreground Color	Current Screen Pixel
1	0	Current Screen Pixel	Cursor Background Color
1	1	NOT Current Screen Pixel	Cursor Foreground Color

The hardware cursor color is taken from the Hardware Graphics Cursor Foreground Stack (CR4A) and the Hardware Graphics Cursor Background Stack (CR4B) registers. Each of these is a stack of three 8-bit registers. The stack pointers are reset to 0 by reading the Hardware Graphics Cursor Mode register (CR45). The color value is then programmed by consecutive writes (low byte, second byte, third byte) to the appropriate (foreground or background) register.

Enabling/Disabling the Cursor

The hardware cursor is disabled when a VGA-compatible mode is in use. It can be enabled or disabled when in Enhanced mode (bit 0 of MM8508 = 1), as follows.

```
CR39 ⇐ A0H          ; Unlock System Control registers
CR45_0 ⇐ 1          ; Enable hardware cursor
CR45_0 ⇐ 0          ; Disable hardware cursor
CR39 ⇐ 00H          ; Lock System Control registers
```

Positioning the Cursor

The cursor can be positioned at any point on the display, with the X,Y coordinates ranging from 0 to 2047. This enables the full cursor images to be displayed on the screen and partial cursor images to be displayed at the right edge and the bottom edge of the screen. The cursor offset OX,OY has to be set to 0,0 for a 1024x768 resolution. If X is > (1024 – 64) or Y is > (768 – 64), then a partial cursor is visible at the right edge or top edge of the screen respectively. Note that if Y ≥ 768 then the cursor is not visible; it is residing in the off-screen area.

A partial cursor image can be displayed at the left edge or the top edge of the screen. To enable partial cursor display at the top edge of the screen, Y is set to 0 and the Y offset register is set to OY (range from 0 to 63). This displays the bottom 64–OY rows of the cursor image at the currently set X position and the top edge of the screen. Similarly, a partial cursor can be displayed at the left edge of the screen by setting X to 0 and the X offset register to OX (range from 0 to 63). This displays the right 64–OX columns of the cursor image at the currently set X and the left edge of the screen. The following pseudocode illustrates cursor positioning.

```
CR39 ⇐ A0H                              ; Unlock System Control registers
CR46_10-8 ⇐ MS 3 bits of X cursor position
CR47_7-0 ⇐ LS 8 bits of X cursor position
```

```
CR48_10-8 ⇐ MS 3 bits of Y cursor position
CR49_7-0 ⇐ LS 8 bits of Y cursor position
CR4E_5-0 ⇐ Cursor Offset X position
CR4F_5-0 ⇐ Cursor Offset Y position
CR39 ⇐ 00H                              ; Lock System Control registers
```

The cursor position is updated by the hardware once each frame. Therefore, the programmer should ensure that the position is re-programmed no more than once for each vertical sync period.

Programming the Cursor Shape

The AND and the XOR cursor image bitmaps are 512 bytes each. These bitmaps are word interleaved in a contiguous area of display memory, i.e., AND word 0, XOR word 0, AND word 1, XOR word 1 ... AND word 255, XOR word 255. The starting location must be on a 1024-byte boundary. This location is programmed into the Hardware Graphics Cursor Start Address registers (CR4C and CR4D) as follows:

```
CR39 ⇐ A0H                              ; Unlock System Control registers
CR4C_11-8 ⇐ MS 4 bits of the cursor storage start 1024-byte segment.
CR4D ⇐ LS 8 bits of the cursor storage start 1024-byte segment
CR39 ⇐ 0                                ; Lock System Control registers
```

The value programmed is the 1024-byte segment of display memory at which the beginning of the hardware cursor bit pattern is located. For example, for an 800x600x8 mode on a 1 MByte system, there are 1024 1K segments. Programming CR4C_11-8 with 3H and CR4D with FEH specifies the starting location as the 1022nd (0-based) 1K segment. The cursor pattern is programmed (using linear addressing) at FF800H offset from the base address of the frame buffer.

Note

If the cursor is not 64 bits by 64 bits, the given images should be padded to make the cursor image 64 bits by 64 bits. The padded area should be made transparent by padding the extra AND mask bits with '1's and the extra XOR bits by '0's.

15.6 BUS MASTER DMA

For PCI systems, ViRGE/VX provides bus master DMA capabilities. There are two independent DMA channels. One handles transfers of video data to video memory or an MPEG decoder and from video memory to system memory. The other is used to transfer command and parameter or image data to the S3d Engine.

15.6.1 Video/Graphics DMA Transfers

These transfers are enabled by setting MM8580_0 to 1. If MM8580_1 = 1, data is transferred from system memory to the LPB output FIFO. This can be compressed video data for transfer to an MPEG decoder or de-compressed software MPEG data to be written to video memory with optional decimation. See the LPB section for the appropriate register settings for each type of transfer. For either case, the starting address in system memory for the data to be transferred is programmed in MM8580_31-2 (doubleword aligned). The number of doublewords to transfer -1 is programmed in MM8584_23-2.

If MM8580_1 = 0, data is transferred from video memory to system memory. The starting address in video memory is programmed in MM8220_21-3 (quadword aligned). The line width in quadwords is programmed in MM8224_27-19 and the line stride in quadwords is programmed in MM8224_11-3. The destination starting address in system memory is programmed in MM8580_31-2 (doubleword aligned). The number of doublewords to transfer -1 is programmed in MM8584_23-2.

15.6.2 S3d Engine Command/Parameter/Image Data DMA Transfers

The type of transfer requires establishment of a locked circular buffer in system memory. MM8590_1 defines this buffer as being 4 or 64 KBytes. The base address for the buffer is programmed in MM8590_31-12 (32K) or 31-16 (64K). S3d Engine DMAs are enabled by setting MM8590_0 to 1.

The DMA write and read pointer registers (MM8594 and MM8598) are initialized to all 0's. The transfer sequence begins with the CPU writing some amount of data to the buffer. This data is derived from the parameter blocks passed to the driver by the application via the programming interface. In general, the transfer should include one or more complete command/parameter/data blocks. After this data is written to the buffer, the next offset address in the frame buffer is programmed into the DMA write pointer field (MM8594_15-0) and MM8594_16 is set to 1 to indicate that the write pointer has been updated. When the write pointer is ahead of the read pointer (MM8598_15-0), DMA transfers to the S3d Engine begin. The read pointer field is automatically updated as each doubleword transfer to the S3d Engine is made. DMA transfers will continue as long as the write pointer is ahead of the read pointer. They stop when the read pointer equals the write pointer.

Additional data can be written to the buffer at any time, starting at the current write pointer address. Wrapping of the writes when the end of the buffer is reached is handled by the programmer. Before writing additional data to the buffer, the programmer must first read the read pointer to determine how much space is available in the buffer. If this is not done, the write data could wrap and overwrite good data before it is read from the buffer.

Each update of the circular buffer must start with a doubleword header that defines what is to follow. The format of this header is:

Bit(s)	Description
15-0	Number of doublewords to transfer
29-16	Most significant 14 bits of the least significant 16 bits of the offset of the first S3d register to be programmed
30	Reserved
31	Data type (0 = register data, 1 = image data)

If image data is being transferred (a BitBLT with the CPU as the source), only bit 31 (=1) and bits 15-0 need be programmed.

This capability allows updating of multiple S3d registers in one DMA operation. For example, defining a color pattern with an 8 bits/pixel color depth requires that all registers from A100H to A13CH be programmed. Thus, bits 15-0 would be programmed with 16 (decimal). The most significant 14 bits of A100H (dropping the two low-order 0's) are programmed into bits 29-16. Bit 31 is cleared to 0.

The parameter register address ranges for some of the commands contain "holes" (no register). The programmer can either send a new header for each contiguous register sequence or program garbage in the doublewords corresponding to the holes. For example, there is a single doubleword gap between the 3AS_RS parameter register for a 3D line and the 3dZ parameter register. This is probably best handled by the "garbage" technique.

Section 16: VGA Standard Register Descriptions

In the following register descriptions, 'U' stands for undefined or unused and 'R' stands for reserved (write = 0, read = U). A question mark in an address stands for a hexadecimal value of either 'B' or 'D'. If bit 0 of the Miscellaneous Output Register (3C2H, Write) is set to 1, the address is based at 3DxH for color emulation. If this bit is reset to 0, the address is based at 3BxH for monochrome emulation.

See Appendix A for a table listing each register in this section and its page number.

16.1 GENERAL REGISTERS

This section describes general input status and output control registers.

Miscellaneous Output Register (MISC)

Write Only — Address: 3C2H
Read Only — Address: 3CCH
Power-On Default: 00H

This register controls miscellaneous output signals. A hardware reset sets all bits to zero.

7	6	5	4	3	2	1	0
VSP	HSP	PGSL	= 0	CLK 1	SEL 0	ENB RAM	IOA SEL

Bit 0 IOA SEL - I/O Address Select
0 = Monochrome emulation. Address based at 3Bx
1 = Color emulation. Address based at 3Dx

Bit 1 ENB RAM - Enable CPU Display Memory Access
0 = Disable access of the display memory from the CPU
1 = Enable access of the display memory from the CPU

Bits 3–2 Clock Select - Select the Video Clock Frequency
00 = Selects the DCLK PLL parameters in SR22 and SR23. The default generates a 25.175 MHz DCLK for 640 horizontal pixels
01 = Selects the DCLK PLL parameters in SR24 and SR25. The default generates a 28.322 MHz DCLK for 720 horizontal pixels
10 = Reserved
11 = Selects the DCLK PLL parameters in SR12 and SR13. This setting is used for all Enhanced modes.

The selected DCLK PLL parameter values are loaded into the PLL when bit 1 of SR15_1 is set to 1 or when SR15_5 is programmed to 1 and then 0.

Bit 4 Reserved = 0

Bit 5 PGSL -Select High 64K Page
0 = Select the low 64K page of memory
1 = Select the high 64K page of memory

Bit 6 $\overline{\text{HSP}}$ - Select Negative Horizontal Sync Pulse
0 = Select a positive horizontal retrace sync pulse
1 = Select a negative horizontal retrace sync pulse

Bit 7 $\overline{\text{VSP}}$ - Select Negative Vertical Sync Pulse
0 = Select a positive vertical retrace sync pulse
1 = Select a negative vertical retrace sync pulse

Feature Control Register (FCR_WT, FCR_AD)

Write Only — Address: 3?AH
Read Only — Address: 3CAH
Power-On Default: 00H

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	VSSL	= 0	= 0	= 0

Bits 2–0 Reserved = 0

Bit 3 VSSL - Vertical Sync Type Select
0 = Enable normal vertical sync output to the monitor
1 = The 'vertical sync' output is the logical OR of 'vertical sync' and 'vertical active display enable' (an internal signal)

Bits 7–4 Reserved = 0

Input Status 0 Register (STATUS_0)

Read Only Address: 3C2H
Power-On Default: Undefined

This register indicates the status of the VGA adapter.

7	6	5	4	3	2	1	0
CRT INTPE	= 0	= 0	MON SENS	= 0	= 0	= 0	= 0

Bits 3–0 Reserved = 0

Bit 4 MON SENS - Monitor Sense Status
0 = The internal $\overline{\text{SENSE}}$ signal is a logical 0
1 = The internal $\overline{\text{SENSE}}$ signal is a logical 1

Bits 6–5 Reserved = 0

Bit 7 CRT INTPE - CRT Interrupt Status
0 = Vertical retrace interrupt cleared
1 = Vertical retrace interrupt pending

See Section 12.7 for an explanation of interrupt generation.

Input Status 1 Register (STATUS_1)

Read Only Address: 3?AH
Power-On Default: Undefined

This register indicates video sync timing and video wraparound.

7	6	5	4	3	2	1	0
		TST-VDT					
= 0	= 0	1	0	VSY	= 1	LPF	DTM

Bit 0 $\overline{\text{DTM}}$ - Display Mode Inactive
0 = The display is in the display mode.
1 = The display is not in the display mode. Either the horizontal or vertical retrace period is active

Bit 1 Reserved = 0

Bit 2 Reserved = 1

Bit 3 VSY - Vertical Sync Active
0 = Display is in the display mode
1 = Display is in the vertical retrace mode

Bits 5–4 TST-VDT - Video Signal Test
Video Data Feedback 1,0. These bits are feedback video signals to do read back tests. These bits are selectively connected to two of the eight color outputs of the attribute controller. Bits 5 and 4 of the color plane enable register (AR12) control the multiplexer for this video output observation.

Bits 7–6 Reserved = 0

Video Subsystem Enable Register

Read/Write Address: 3C3H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	VGA ENB

Bit 0 VGA ENB - VGA Enable
0 = VGA display disabled
1 = VGA display enabled

Bits 7–1 Reserved

16.2 SEQUENCER REGISTERS

The sequencer registers are located at two-byte address spaces. These registers are accessed by first writing the data to the index register of the sequencer at I/O address 3C4H and then writing to or reading from the data register at 3C5H.

Sequencer Index Register (SEQX)

Read/Write Address: 3C4H
Power-On Default: Undefined

This register is loaded with a binary value that indexes the sequencer register for read/write data. This value is referred to as the "Index Number" of the SR register in this document.

7	6	5	4	3	2	1	0
R	R	R	SEQ ADDRESS				

Bits 4–0 SEQ ADDRESS - Sequencer Register Index
A binary value indexing the register where data is to be accessed.

Bits 7–5 Reserved

Sequencer Data Register (SEQ_DATA)

Read/Write Address: 3C5H
Power-On Default: Undefined

This register is the data port for the sequencer register indexed by the Sequencer Index register (3C4H).

7	6	5	4	3	2	1	0
SEQ DATA							

Bit 7–0 SEQ DATA - Sequencer Register Data
Data to the sequencer register indexed by the sequencer address index.

Reset Register (RST_SYNC) (SR0)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 00H

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	= 0	= 0	SYN RST	ASY RST

Bit 0 $\overline{\text{ASY}}$ RST - Asynchronous Reset

This bit is for VGA software compatibility only. It has no function for ViRGE.

Bit 1 $\overline{\text{SYN}}$ RST - Synchronous Reset

This bit is for VGA software compatibility only. It has no function for ViRGE.

Bits 7–2 Reserved = 0

Clocking Mode Register (CLK_MODE) (SR1)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 01H

This register controls the operation mode of dot clock and character clock.

7	6	5	4	3	2	1	0
= 0	= 0	SCRN OFF	SHF 4	DCK 1/2	SHF LD	= 0	8DC

Bit 0 8DC - 8 Dot Clock Select
0 = Character clocks 9 dots wide are generated
1 = Character clocks 8 dots wide are generated

Bit 1 Reserved = 0

Bit 2 SHF LD - Load Serializers Every Second Character Clock
0 = Load the video serializer every character clock
1 = Load the video serializers every other character clock

Bit 3 DCK 1/2 - Internal Dot Clock = 1/2 DCLK
0 = Set the internal dot clock to the same frequency as DCLK
1 = Set the internal dot clock to 1/2 the frequency of DCLK

This bit is used for horizontal pixel doubling.

Bit 4 SHF 4 - Load Serializers Every Fourth Character Clock
0 = Load the serializers every character clock cycle
1 = Load the serializers every fourth character clock cycle

Bit 5 SCRN OFF - Screen Off
0 = Screen is turned on.
1 = Screen is turned off

Bit 7–6 Reserved = 0

Enable Write Plane Register (EN_WT_PL) (SR2)

Read/Write Address: 3C5H, Index 02H
Power-On Default: 00H

This register selects write protection or write permission for CPU write access into video memory.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	EN.WT.PL.			

Bits 3–0 EN.WT.PL - Enable Write to a Plane
0 = Disables writing into the corresponding plane
1 = Enables the CPU to write to the corresponding color plane

Bits 7–4 Reserved = 0

Character Font Select Register (CH_FONT_SL) (SR3)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 03H

7	6	5	4	3	2	1	0
= 0	= 0	SLA 2	SLB 2	SLA 1	0	SLB 1	0

In text modes, bit 3 of the attribute byte normally turns the foreground intensity on or off. This bit can be redefined to be a switch between two character sets. The switch is enabled when there is a difference between the value of character font select A and character font select B bits. Memory Mode (SR4) register bit 1 = 1 (extended memory) enables all bits of this function; otherwise character fonts 0 and 4 are available. 256 KBytes of video memory support 8 character sets. This register is reset to 0 asynchronously during a system reset.

Bits 4, 1–0 SLB - Select Font B
This value selects the portion of plane 2 used to generate text character fonts when bit 3 of the attribute byte is a logical 1, according to the following table:

Bits 4,1,0	Font Table Location	Bits 4, 1,0	Font Table Location
000	First 8K of plane 2	100	Second 8K of plane 2
001	Third 8K of plane 2	101	Fourth 8K of plane 2
010	Fifth 8K of plane 2	110	Sixth 8K of plane 2
011	Seventh 8K of plane 2	111	Eighth 8K of plane 2

Bits 5, 3–2 SLA - Select Font A
This value selects the portion of plane 2 used to generate text character fonts when bit 3 of attribute byte is a logical 0, according to the same table as the character font select A.

Bits 7–6 Reserved = 0

Memory Mode Control Register (MEM_MODE) (SR4)

Read/Write Address: 3C5H, Index 04H
Power-On Default: 00H

This register controls CPU memory addressing mode.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	CHN 4M	SEQ MODE	EXT MEM	= 0

Bit 0 Reserved = 0

Bit 1 EXT MEM - Extended Memory Access
0 = Memory access restricted to 16/32 KBytes
1 = Allows complete memory access to 256 KBytes. Required for VGA

Bit 2 SEQ MODE - Sequential Addressing Mode
This bit affects only CPU write data accesses into video memory. Bit 3 of this register must be 0 for this bit to be effective.
0 = Enables the odd/even addressing mode. Even addresses access planes 0 and 2. Odd addresses access planes 1 and 3
1 = Directs the system to use a sequential addressing mode

Bit 3 CHN 4M - Select Chain 4 Mode
0 = Enables odd/even mode.
1 = Chain 4 Mode. This bit selects modulo 4 addressing for CPU access to display memory. A logical 1 directs the two lower order bits of the CPU address used to select the plane in video memory to be accessed as follows:

A1	A0	Plane Selected
0	0	0
0	1	1
1	0	2
1	1	3

Bits 7–4 Reserved = 0

16.3 CRT CONTROLLER REGISTERS

The CRT controller registers are located at two locations in I/O address space. These registers are accessed by first writing to the index register of the CRT controller and then accessing the data register. The index register is located at I/O address 3?4H and the CRT Controller Data register is at 3?5H. Which address is used (3BX or 3DX) depends on bit 0 of the Miscellaneous Output register at 3C2H.

CRT Controller Index Register (CRTC_ADR) (CRX)

Read/Write Address: 3?4H
Power-On Default: 00H

This register is loaded with a binary value that indexes the CRT controller register where data is to be accessed. This value is referred to as the "Index Number" of the CR register (CR00–18). This register is also used as an index to the S3 VGA registers, the System Control Registers and the System Extension registers.

7	6	5	4	3	2	1	0
CRTC ADDRESS							

Bits 7–0 CRTC ADDRESS - CRTC Register Index
A binary value indexing the register where data is to be accessed.

CRT Controller Data Register (CRTC_DATA) (CRT)

Read/Write Address: 3?5H
Power-On Default: Undefined

This register is the data port for the CRT controller register indexed by the CRT Controller Address register.

7	6	5	4	3	2	1	0
CRTC DATA							

Bits 7–0 CRTC DATA - CRTC Register Data
Data to the CRT controller register indexed by the CRT controller address index.

Horizontal Total Register (H_TOTAL) (CR0)

Read/Write Address: 3?5H, Index 00H
Power-On Default: Undefined

This register defines the number of character clocks from HSYNC going active to the next HSYNC going active. In other words, it is the total time required for both the displayed and non-displayed portions of a single scan line. Bit 8 of this value is bit 0 of CR5D.

7	6	5	4	3	2	1	0
HORIZONTAL TOTAL							

Bits 7–0 HORIZONTAL TOTAL.
9-bit Value = (number of character clocks in one scan line) - 5. This register contains the least significant 8 bits of this value.

Horizontal Display End Register (H_D_END) (CR1)

Read/Write Address: 3?5H, Index 01H
Power-On Default: Undefined

This register defines the number of character clocks for one line of the active display. Bit 8 of this value is bit 1 of CR5D.

7	6	5	4	3	2	1	0
HORIZONTAL DISPLAY END							

Bits 7–0 HORIZONTAL DISPLAY END
9-bit Value = (number of character clocks of active display) - 1. This register contains the least significant 8 bits of this value.

Start Horizontal Blank Register (S_H_BLNK) (CR2)

Read/Write Address: 3?5H, Index 02H
Power-On Default: Undefined

This register specifies the value of the character clock counter at which the $\overline{\text{BLANK}}$ signal is asserted. Bit 8 of this value is bit 2 of CR5D.

7	6	5	4	3	2	1	0
START HORIZONTAL BLANK							

Bits 7–0 START HORIZONTAL BLANK
9-bit Value = character clock value at which horizontal blanking begins. This register contains the least significant 8 bits of this value.

End Horizontal Blank Register (E_H_BLNK) (CR3)

Read/Write Address: 3?5H, Index 03H
Power-On Default: Undefined

This register determines the pulse width of the $\overline{\text{BLANK}}$ signal and the display enable skew.

7	6	5	4	3	2	1	0
R	DSP-SKW 1	0	END HORIZONTAL BLANK				

Bits 4–0 END HORIZONTAL BLANK
7-bit Value = least significant 7 bits of the character clock counter value at which time horizontal blanking ends. To obtain this value, add the desired $\overline{\text{BLANK}}$ pulse width in character clocks to the Start Horizontal Blank value, which is also in character clocks. The 5 least significant bits of this sum are programmed into this field. The sixth bit is programmed into bit 7 of CR5. The seventh bit is programmed into bit 3 of CR5D.

Bits 6–5 DSP-SKW - Display Skew
These two bits determine the amount of display enable skew. Display enable skew control provides sufficient time for the CRT Controller to access the display buffer to obtain a character and attribute code, access the character generator font, and then go through the Horizontal Pixel Panning register in the Attribute Controller. Each access requires the display enable signal to be skewed one character clock unit so the video output is synchronous with the HSYNC and VSYNC signals. The bit values and amount of skew are shown in the following table:
00 = Zero character clock skew
01 = One character clock skew
10 = Two character clock skew
11 = Three character clock skew

Bit 7 Reserved

Start Horizontal Sync Position Register (S_H_SY_P) (CR4)

Read/Write Address: 3?5H, Index 04H
Power-On Default: Undefined

This register is used to adjust the screen center horizontally and to specify the character position at which HSYNC becomes active. Bit 8 of this value is bit 4 of CR5D.

7	6	5	4	3	2	1	0
START HORIZONTAL SYNC POSITION							

Bits 7–0 START HORIZONTAL SYNC POSITION.
9-bit Value = character clock counter value at which HSYNC becomes active. This register contains the least significant 8 bits of this value.

End Horizontal Sync Position Register (E_H_SY_P) (CR5)

Read/Write Address: 3?5H, Index 05H
Power-On Default: Undefined

This register specifies when the HSYNC signal becomes inactive and the horizontal skew. The HSYNC pulse defined by this register can be extended by 32 DCLKs via bit 5 of CR5D.

7	6	5	4	3	2	1	0
EHB b5	HOR-SKW 1	0	END HORIZONTAL SYNC POS				

Bits 4–0 END HORIZONTAL SYNC POS
6-bit Value = 6 least significant bits of the character clock counter value at which time HSYNC becomes inactive. To obtain this value, add the desired HSYNC pulse width in character clocks to the Start Horizontal Sync Position value, also in character clocks. The 5 least significant bits of this sum are programmed into this field. The sixth bit is programmed into bit 5 of CR5D.

Bits 6–5 HOR-SKW - Horizontal Skew
These bits control the skew of the HSYNC signal. A binary 00 equals no HSYNC delay. For some modes, it is necessary to provide an HSYNC signal that takes up the entire blanking interval. Some internal timings are generated by the falling edge of the HSYNC signal. To guarantee the signals are latched properly, HSYNC is asserted before the end of the display enable signal, and then skewed several character clock times to provide the proper screen centering.
00 = Zero character clock skew
01 = One character clock skew
10 = Two character clock skew
11 = Three character clock skew

Bit 7 EHB b5
End Horizontal Blanking bit 5.

Vertical Total Register (V_TOTAL) (CR6)

Read/Write Address: 3?5H, Index 06H
Power-On Default: Undefined

This register specifies the number of scan lines from one VSYNC active to the next VSYNC active. The scan line counter resets to 0 at this point. Bit 8 is bit 0 of CR7. Bit 9 is bit 5 of CR7. Bit 10 is bit 0 of CR5E.

7	6	5	4	3	2	1	0
VERTICAL TOTAL							

Bits 7–0 VERTICAL TOTAL
11-bit Value = (number of scan lines from VSYNC active to the next VSYNC active) - 2. This register contains the least significant 8 bits of this value.

CRTC Overflow Register (OVFL_REG) (CR7)

Read/Write Address: 3?5H, Index 07H
Power-On Default: Undefined

7	6	5	4	3	2	1	0
VRS 9	VDE 9	VT 9	LCM 8	SVB 8	VRS 8	VDE 8	VT 8

This register provides extension bits for fields in other registers.

Bit 0 Bit 8 of the Vertical Total register (CR6)

Bit 1 Bit 8 of the Vertical Display End register (CR12)

Bit 2 Bit 8 of the Vertical Retrace Start register (CR10)

Bit 3 Bit 8 of the Start Vertical Blank register (CR15)

Bit 4 Bit 8 of the Line Compare register (CR18)

Bit 5 Bit 9 of the Vertical Total register (CR6)

Bit 6 Bit 9 of the Vertical Display End register (CR12)

Bit 7 Bit 9 of the Vertical Retrace Start register (CR10)

Preset Row Scan Register (P_R_SCAN) (CR8)

Read/Write Address: 3?5H, Index 08H
Power-On Default: Undefined

This register is used for the pixel scrolling and panning, and text formatting and vertical scrolling.

7	6	5	4	3	2	1	0
= 0	BYTE-PAN 1	0	PRE-SET ROW SCAN COUNT				

Bits 4–0 PRE-SET ROW SCAN COUNT
Value = starting row within a character cell for the first character row displayed after vertical retrace. This allows a partial character row to be displayed at the top of the display and is used for scrolling.

Bits 6–5 BYTE-PAN
Value = number of bytes to pan. The number of pixels to pan is specified in AR13.

Bit 7 Reserved = 0

Maximum Scan Line Register (MAX_S_LN) (CR9)

Read/Write Address: 3?5H, Index 09H
Power-On Default: Undefined

This register specifies the number of scan lines per character row and provides one scanning control bit and two overflow bits.

7	6	5	4	3	2	1	0
DBL SCN	LCM 9	SVB 9	MAX SCAN LINE				

Bits 4–0 MAX SCAN LINE
Value = (number of scan lines per character row) - 1

Bit 5 SVB 9
Bit 9 of the Start Vertical Blank Register (CR15)

Bit 6 LCM 9
Bit 9 of the Line Compare Register (CR18)

Bit 7 DBL SCN
0 = Normal operation
1 = Enables double scanning operation. Each line is displayed twice by repeating the row scan counter and video memory address. Vertical parameters in the CRT controller are not affected.

Cursor Start Scan Line Register (CSSL) (CRA)

Read/Write Address: 3?5H, Index 0AH
Power-On Default: Undefined

The cursor start register defines the row scan of a character line where the cursor begins.

7	6	5	4	3	2	1	0
= 0	= 0	CSR OFF	CSR CURSOR START SCAN LINE				

Bits 4–0 CSR CURSOR START SCAN LINE
Value = (starting cursor row within the character cell) - 1. When the cursor start register is programmed with a value greater than the cursor end register, no cursor is generated.

Bit 5 CSR OFF
0 = Turns on the text cursor
1 = Turns off the text cursor

Bits 7–6 Reserved = 0

Cursor End Scan Line Register (CESL) (CRB)

Read/Write Address: 3?5H, Index 0BH
Power-On Default: Undefined

This register defines the row scan of a character line where the cursor ends.

7	6	5	4	3	2	1	0
= 0	CSR-SKW 1	0	CURSOR END SCAN LINE				

Bits 4–0 CURSOR END SCAN LINE
Value = ending scan line number within the character cell for the text cursor. If the value of the cursor start scan line is greater than the value of cursor end line, then no cursor is generated.

Bits 6–5 CSR-SKW - Cursor Skew
These bits control the delay skew of the cursor signal. Cursor skew delays the text cursor by the selected number of clocks. For example, a skew of 1 moves the cursor right one character position on the screen.
00 = Zero character clock skew
01 = One character clock skew
10 = Two character clock skew
11 = Three character clock skew

Bit 7 Reserved = 0

Start Address High Register (STA(H)) (CRC)

Read/Write Address: 3?5H, Index 0CH
Power-On Default: Undefined

15	14	13	12	11	10	9	8
DISPLAY START ADDRESS (HIGH)							

21-bit Value = the first address after a vertical retrace at which the display on the screen begins on each screen refresh. These along with bits 4-0 of CR69 are the high order start address bits.

Start Address Low Register (STA(L)) (CRD)

Read/Write Address: 3?5H, Index 0DH
Power-On Default: Undefined

7	6	5	4	3	2	1	0
DISPLAY START ADDRESS (LOW)							

Start address (low) contains the 8 low order bits of the address.

Cursor Location Address High Register (CLA(H)) (CRE)

Read/Write Address: 3?5H, Index 0EH
Power-On Default: Undefined

15	14	13	12	11	10	9	8
CURSOR LOCATION ADDRESS (HIGH)							

21-bit Value = the cursor location address of the video memory where the text cursor is active. This register along with bits 4-0 of CR69 are the high order bits of the address.

Cursor Location Address Low Register (CLA(L)) (CRF)

Read/Write Address: 3?5H, Index 0FH
Power-On Default: Undefined

7	6	5	4	3	2	1	0
CURSOR LOCATION ADDRESS (LOW)							

Cursor location address (low) contains the 8 low order bits of the address.

Vertical Retrace Start Register (VRS) (CR10)

Read/Write Address: 3?5H, Index 10H
Power-On Default: Undefined

7	6	5	4	3	2	1	0
VERTICAL RETRACE START							

Bits 7–0 VERTICAL RETRACE START.
11-bit Value = scan line counter value at which VSYNC becomes active. These are the low-order 8 bits. Bit 8 is bit 2 of CR7. Bit 9 is bit 7 of CR7. Bit 10 is bit 4 of CR5E.

Vertical Retrace End Register (VRE) (CR11)

Read/Write Address: 3?5H, Index 11H
Power-On Default: 0xH

This register controls the vertical interrupt and CR0-7

7	6	5	4	3	2	1	0
LOCK R0-7	REF 3/5	DIS VINT	CLR VINT	VERTICAL RETRACE END			

Bits 3–0 VERTICAL RETRACE END
Value = least significant 4 bits of the scan line counter value at which VSYNC goes inactive. To obtain this value, add the desired VSYNC pulse width in scan line units to the CR10 value, also in scan line units. The 4 least significant bits of this sum are programmed into this field. This allows a maximum VSYNC pulse width of 15 scan line units.

Bit 4 $\overline{\text{CLR}}$ VINT - Clear Vertical Retrace Interrupt
0 = Vertical retrace interrupt cleared
1 = The flip-flop is able to catch the next interrupt request

At the end of active vertical display time, a flip-flop is set for a vertical interrupt. The output of this flip-flop goes to the system interrupt controller. The CPU has to reset this flip-flop by writing a logical 0 to this bit while in the interrupt process, then set the bit to 1 to allow the flip-flop to catch the next interrupt request. Do not change the other bits in this register. This bit is cleared to 0 by the BIOS during a mode set, a reset, or power-on.

Bit 5 DIS VINT - Disable Vertical Interrupt
0 = Vertical retrace interrupt enabled if CR32_4 = 1
1 = Vertical interrupt disabled. This bit is cleared to 0 by the BIOS during a mode set, a reset, or power-on

Bit 6 REF $\overline{3}$/5 - Refresh Cycle Select
0 = Three DRAM refresh cycles generated per horizontal line
1 = Five DRAM refresh cycles generated per horizontal line. Selecting five refresh cycles allows use of the VGA chip with slow sweep rate displays (15.75 KHz). This bit is cleared to 0 by the BIOS during a mode set, a reset, or power-on. This setting can be overridden via bits 1-0 of CR3A

Bit 7 LOCK R0-7 - Lock Writes to CRT Controller Registers
0 = Writing to all CRT Controller registers enabled
1 = Writing to all bits of the CRT Controller registers CR0–CR7 except bit 4 of CR7 (LCM8) disabled. This bit is set to 1 by the BIOS during a mode set, a reset or power-on

Vertical Display End Register (VDE) (CR12)

Read/Write Address: 3?5H, Index 12H
Power-On Default: Undefined

The vertical display enable end register defines 8 bits of the 10-bit address of the scan line where the display on the screen ends. Bit 8 and Bit 9 are bits 1 and 6 of CR7. Bit 10 is bit 1 of CR5E.

7	6	5	4	3	2	1	0
VERTICAL DISPLAY END							

Bit 7–0 VERTICAL DISPLAY END
11-bit Value = (number of scan lines of active display) - 1. This register contains the least significant 8 bits of this value.

Offset Register (SCREEN-OFFSET) (CR13)

Read/Write Address: 3?5H, Index 13H
Power-On Default: Undefined

This register specifies the logical line width of the screen and is sometimes called the screen pitch. The starting memory address for the next display row is larger than the current row by two, four or eight times this amount. Bits 5-4 of CR51 are extension bits 9-8 of this register. If these bits are 00b, bit 2 of CR43 is extension bit 8 of this register.

7	6	5	4	3	2	1	0
LOGICAL SCREEN WIDTH							

Bits 7–0 LOGICAL SCREEN WIDTH
10-bit Value = quantity that is multiplied by 2 (word mode), 4 (doubleword mode) or 8 (quadword mode) to specify the difference between the starting byte addresses of two consecutive scan lines. This register contains the least significant 8 bits of this value. The addressing mode is specified by bit 6 of CR14 and bit 3 of CR17. Setting bit 3 of CR31 to 1 forces doubleword mode.

Underline Location Register (ULL) (CR14)

Read/Write Address: 3?5H, Index 14H
Power-On Default: Undefined

This register specifies the horizontal row scan position of underline and display buffer addressing modes.

7	6	5	4	3	2	1	0
= 0	DBWD MODE	CNT BY4	UNDER LINE LOCATION				

Bits 4–0 UNDER LINE LOCATION
5-bit Value = (scan line count of a character row on which an underline occurs) -1

Bit 5 CNT BY4 - Select Count by 4 Mode
0 = The memory address counter depends on bit 3 of CR17 (count by 2)
1 = The memory address counter is incremented every four character clocks

The CNT BY4 bit is used when double word addresses are used.

Bit 6 DBLWD MODE - Select Doubleword Mode
0 = The memory addresses are byte or word addresses
1 = The memory addresses are doubleword addresses

Bit 7 Reserved = 0

Start Vertical Blank Register (SVB) (CR15)

Read/Write Address: 3?5H, Index 15H
Power-On Default: Undefined

This register specifies the scan line at which the vertical blanking period begins. Bit 8 is bit 3 of CR7. Bit 9 is bit 5 of CR9. Bit 10 is bit 2 of CR5E.

7	6	5	4	3	2	1	0
START VERTICAL BLANK							

Bits 7–0 START VERTICAL BLANK.
11-bit va'ue = (scan line count at which $\overline{\text{BLANK}}$ becomes active) - 1. This register contains the least significant 8 bits of this value.

End Vertical Blank Register (EVB) (CR16)

Read/Write Address: 3?5H, Index 16H
Power-On Default: Undefined

This register specifies the scan line count value when the vertical blank period ends.

7	6	5	4	3	2	1	0
END VERTICAL BLANK							

Bits 7–0 END VERTICAL BLANK
Value = least significant 8 bits of the scan line counter value at which vertical blanking ends. To obtain this value, add the desired width of the vertical blanking pulse in scan lines to [(value in the Start Vertical Blank register)-1], also in scan lines. The 8 least significant bits of this sum are programmed into this field. This allows a maximum vertical blanking pulse of 255 scan line units.

CRTC Mode Control Register (CRT_MD) (CR17)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 17H

This register is a multifunction control register, with each bit defining a different specification.

7	6	5	4	3	2	1	0
RST	BYTE MODE	ADW 16K	= 0	WRD MODE	VT X2	4BK HGC	2BK CGA

Bit 0 $\overline{2BK}$ CGA - Select Bank 2 Mode for CGA Emulation
0 = Row scan counter bit 0 is substituted for memory address bit 13 during active display time
1 = Memory address bit 13 appears on the memory address output bit 13 signal of the CRT controller

This bit allows memory mapping compatibility with the IBM CGA graphics mode.

Bit 1 $\overline{4BK}$ HGC - Select Bank 4 Mode for HGA Emulation
0 = Row scan counter bit 1 is substituted for memory address bit 14 during active display time
1 = Memory address bit 14 appears on the memory address output bit 14 signal of the CRT controller

The combination of this bit and bit 0 of this register allows compatibility with Hercules HGC graphics memory mapping.

Bit 2 VT X2 - Select Vertical Total Double Mode
0 = Horizontal retrace clock selected
1 = Horizontal retrace clock divided by two selected

This bit selects horizontal retrace clock or horizontal retrace clock divided by two as the clock that controls the vertical timing counter. If the vertical retrace counter is clocked with the horizontal retrace clock divided by 2, then the vertical resolution is double.

Bit 3 CNT BY2 - Select Word Mode
0 = Memory address counter is clocked with the character clock input, and byte mode addressing for the video memory is selected
1 = Memory address counter is clocked by the character clock input divided by 2, and word mode addressing for the video memory is selected

Bit 4 Reserved = 0

Bit 5 $\overline{\text{ADW}}$ 16K - Address Wrap
0 = When word mode is selected by bit 6 of this register, memory address counter bit 13 appears on the memory address output bit 0 signal of the CRT controller and the video memory address wraps around at 16 KBytes
1 = When word mode is selected by bit 6 of this register, memory address counter bit 15 appears on the memory address output bit 0 signal of the CRT controller

This bit is useful in implementing IBM CGA mode.

Bit 6 BYTE MODE - Select Byte Addressing Mode
0 = Word mode shifts all memory address counter bits down one bit, and the most significant bit of the counter appears on the least significant bit of the memory address output
1 = Byte address mode

Bit 7 $\overline{\text{RST}}$ - Hardware Reset
0 = Vertical and horizontal retrace pulses always inactive
1 = Vertical and horizontal retrace pulses enabled

This bit does not reset any other registers or outputs.

Line Compare Register (LCM) (CR18)

Read/Write Address: 3?5H, Index 18H
Power-On Default: Undefined

This register is used to implement a split screen function. When the scan line counter value is equal to the content of this register, the memory address counter is cleared to 0. The linear address counter then sequentially addresses the display buffer starting at address 0. Each subsequent row address is determined by the addition of the Offset (CR13) register content. Bit 8 is bit 4 of CR7. Bit 9 is bit 6 of CR9. Bit 10 is bit 6 of CR5E.

7	6	5	4	3	2	1	0
LINE COMPARE POSITION							

Bit 7–0 LINE COMPARE POSITION
11-bit Value = number of scan lines at which the screen is split into screen A and screen B. This register contains the least significant 8 bits of this value.

CPU Latch Data Register (GCCL) (CR22)

Read Only Address: 3?5H, Index 22H
Power-On Default: Undefined

This register is used to read the CPU latch in the Graphics Controller.

7	6	5	4	3	2	1	0
GRAPHICS CONTROLLER CPU LATCH - N							

Bits 7–0 GRAPHICS CONTROLLER CPU LATCH - N
Bits 1-0 of GR4 select the latch number N (3-0) of the CPU Latch.

Attribute Index Register (ATC_F/I) (CR24)

Read Only Address: 3?5H, Index 24H, 26H
Power-On Default: Undefined

This register is used to read the value of the Attribute Controller Index register and its associated internal address flip-flop (AFF). It can be read at either index 24H or 26H.

7	6	5	4	3	2	1	0
AFF	= 0	ENV	ATTRIBUTE CONTROLLER INDEX				

Bits 4–0 ATTRIBUTE CONTROLLER INDEX
This value is the Attribute Controller Index Data at I/O port 3C0H.

Bit 5 ENV- Enable Video Display
This is the setting of bit 5 of 3C0H, indicating video display enabled status (1 = enabled).

Bit 6 Reserved = 0

Bit 7 $\overline{\text{AFF}}$
Inverted Internal Address flip-flop

16.4 GRAPHICS CONTROLLER REGISTERS

The graphics controller registers are located at a two byte I/O address space. These registers are accessed by first writing an index to the Graphics Address register (at 3CEH) and then accessing the Data register (at 3CFH).

Graphics Controller Index Register (GRC_ADR)

Read/Write Address: 3CEH
Power-On Default: Undefined

This register is loaded with a binary index value that determines which graphics controller register will be accessed. This value is referred to as the "Index Number" of the GR register (GR0–6).

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	GR CONT ADDRESS			

Bits 3–0 GR CONT ADDRESS - Graphics Controller Register Index
A binary value indexing the register where data is to be accessed.

Bits 7–4 Reserved = 0

Graphics Controller Data Register (GRC_DATA)

Read/Write Address: 3CFH
Power-On Default: Undefined

This register is the data port for the graphics controller register indexed by the Graphics Controller Index register.

7	6	5	4	3	2	1	0
GRAPHICS CONTROLLER DATA							

Bit 7–0 GRAPHICS CONTROLLER DATA
Data to the Graphics Controller register indexed by the graphics controller address.

Set/Reset Data Register (SET/RST_DT) (GR0)

Read/Write Address: 3CFH, Index 00H
Power-On Default: Undefined

This register represents the value written to all 8 bits of the respective memory plane when the CPU executes a memory write in write modes 0 and 3.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	SET/RESET DATA			

Bits 3–0 SET/RESET DATA
These bits become the color value for CPU memory write operations. In write mode 0, the set/reset data can be enabled on the corresponding bit of the Enable Set/Reset Data register. In write mode 3, there is no effect on the Enable Set/Reset Data register.

Bits 7–4 Reserved = 0

Enable Set/Reset Data Register (EN_S/R_DT) (GR1)

Read/Write Address: 3CFH, Index 01H
Power-On Default: Undefined

These bits enable the set/reset data, and affect write mode 0.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	ENB SET/RST DATA			

Bits 3–0 ENB SET/RST DATA
When each bit is a logical 1, the respective memory plane is written with the value of the Set/Reset Data register. A logical 0 disables the set/reset data in a plane, and that plane is written with the value of CPU write data.

Bits 7–4 Reserved = 0

Color Compare Register (COLOR-CMP) (GR2)

Read/Write Address: 3CFH, Index 02H
Power-On Default: Undefined

These bits represent a 4-bit color value to be compared. In read mode 1, the CPU executes a memory read, the read data is compared with this value and returns the results. This register works in conjunction with the Color Don't Care register.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	COLOR COMPARE DATA			

Bits 3–0 COLOR COMPARE DATA
This value becomes the reference color used to compare each pixel. Each of the 8-bit positions of the read data are compared across four planes and a logical 1 is returned in each bit position for which the colors match.

Bits 7–4 Reserved = 0

Raster Operation/Rotate Count Register (WT_ROP/RTC) (GR3)

Read/Write Address: 3CFH, Index 03H
Power-On Default: Undefined

This register selects a raster operation function and indicates the number of bits the CPU data will be rotated (right) on the video memory write operation.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	RST-OP 1	0	ROTATE-COUNT		

Bits 2–0 ROTATE-COUNT
These bits define a binary encoded value of the number of positions to right-rotate data during a CPU memory write. To write non-rotated data, the CPU must preset a count of 0.

Bits 4–3 RST-OP - Select Raster Operation
The data written to memory can operate logically with the data already in the processor latches. This function is not available in write mode 1. The logical functions are defined as follows:
00 = No operation
01 = Logical AND with latched data
10 = Logical OR with latched data
11 = Logical XOR with latched data

The logical function specified by this register is applied to data being written to memory while in modes 0, 2 and 3.

Bits 7–5 Reserved = 0

Read Plane Select Register (RD_PL_SL) (GR4)

Read/Write Address: 3CFH, Index 04H
Power-On Default: Undefined

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	= 0	= 0	RD-PL-SL 1	0

The contents of this register represent the memory plane from which the CPU reads data in read mode 0. This register has no effect on the color compare read mode (read mode 1). In odd/even mode, bit 0 is ignored. Four memory planes are selected as follows:

Bits 1–0 RD-PL-SL - Read Plane Select
The memory plane is selected as follows:
00 = Plane 0
01 = Plane 1
10 = Plane 2
11 = Plane 3

Bits 7–2 Reserved = 0

Graphics Controller Mode Register (GRP_MODE) (GR5)

Read/Write Address: 3CFH, Index 05H
Power-On Default: Undefined

7	6	5	4	3	2	1	0
	SHF-MODE		O/E	RD		WRT-MD	
= 0	256	O/E	MAP	CMP	= 0	1	0

This register controls the mode of the Graphics Controller as follows:

Bit 1–0 WRT-MD - Select Write Mode
These bits select the CPU write mode into video memory. The function of each mode is defined as follows:
00 = Write Mode 0. Each of four video memory planes is written with the CPU data rotated by the number of counts in the rotate register. If the Set/Reset register is enabled for any of four planes, the corresponding plane is written with the data stored in the set/reset register. Raster operations and bit mask registers are effective
01 = Write Mode 1. Each of four video memory planes is written with the data in the processor latches. These latches are loaded during previous CPU read operations. Raster operation, rotate count, set/reset data, enable set/reset data and bit mask registers are not effective
10 = Write Mode 2. Memory planes 0-3 are filled with 8 bits of the value of CPU write data bits 0-3, respectively. For example, if write data bit 0 is a 1, eight 1's are written to memory plane 0. The data on the CPU data bus is treated as the color value. The Bit Mask register is effective as the Mask register. A logical 1 in the Bit Mask register sets the corresponding pixel in the addressed byte to the color specified on the data bus. A logical 0 in the Bit Mask register sets the corresponding pixel in the addressed byte to the corresponding pixel in the processor latches. The Set/Reset, Enable Set/Reset and Rotate Count registers are ignored
11 = Write Mode 3. Each of four video memory planes is written with 8 bits of the color value contained in the set/reset register for that plane. The Enable Set/Reset register is not effective. Rotated CPU write data is ANDed with the bit mask register to form an 8-bit value that performs the same function as the Bit Mask register in write modes 0 and 2. This write mode can be used to fill an area with a single color and pattern

Bit 2 Reserved = 0

Bit 3 RD CMP - Enable Read Compare
0 = The CPU reads data from the video memory planes. The plane is selected by the Read Plane Select register. This is called read mode 0
1 = The CPU reads the results of the logical comparison between the data in four video memory planes selected by the contents of the Color Don't Care register and the contents of the Color Compare register. The result is a 1 for a match and 0 for a mismatch on each pixel. This is called read mode 1

Bit 4 O/E MAP - Select Odd/Even Addressing
0 = Standard addressing.
1 = Odd/even addressing mode selected. Even CPU addresses access plane 0 and 2, while odd CPU addresses access plane 1 and 3. This option is useful for emulating the CGA compatible mode. The value of this bit should be the inverted value programmed in bit 2 of the Sequencer Memory Mode register (SR4). This bit affects reading of display memory by the CPU

Bit 5 SHF-MODE - Select Odd/Even Shift Mode
0 = Normal shift mode
1 = The video shift registers in the graphics section are directed to format the serial data stream with even-numbered bits from both planes on the even-numbered planes and odd-numbered bits from both planes on the odd planes

Bit 6 SHF-MODE - Select 256 Color Shift Mode
0 = Bit 5 in this register controls operation of the video shift registers
1 = The shift registers are loaded in a manner that supports the 256 color mode

Bit 7 Reserved = 0

Memory Map Mode Control Register (MISC_GM) (GR6)

Read/Write Address: 3CFH, Index 06H
Power-On Default: Undefined

This register controls the video memory addressing.

7	6	5	4	3	2	1	0
				MEM-MAP		CHN	TXT
= 0	= 0	= 0	= 0	1	0	O/E	/GR

Bit 0 $\overline{\text{TXT}}$/GR - Select Text/Graphics Mode
0 = Text mode display addressing selected
1 = Graphics mode display addressing selected. When set to graphics mode, the character generator address latches are disabled

Bit 1 CHN O/E - Chain Odd/Even Planes
0 = A0 address bit unchanged
1 = CPU address bit A0 is replaced by a higher order address bit. The content of A0 determines which memory plane is to be addressed. A0 = 0 selects planes 0 and 2, and A0 = 1 selects planes 1 and 3. This mode can be used to double the address space into video memory

Bits 3–2 MEM-MAP - Memory Map Mode
These bits control the address mapping of video memory into the CPU address space. The bit functions are defined below.
00 = A0000H to BFFFFH (128 KBytes)
01 = A0000H to AFFFFH (64 KBytes)
10 = B0000H to B7FFFH (32 KBytes)
11 = B8000H to BFFFFH (32 KBytes)

Bits 7–4 Reserved = 0

Color Don't Care Register (CMP_DNTC) (GR7)

Read/Write Address: 3CFH, Index 07H
Power-On Default: Undefined

This register is effective in read mode 1, and controls whether the corresponding bit of the Color Compare Register is to be ignored or used for color comparison.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	COMPARE PLANE SEL			

Bits 3–0 COMPARE PLANE SEL - Compare Plane Select
0 = The corresponding color plane becomes a don't care plane when the CPU read from the video memory is performed in read mode 1
1 = The corresponding color plane is used for color comparison with the data in the Color Compare register

Bits 7–4 Reserved = 0

Bit Mask Register (BIT_MASK) (GR8)

Read/Write Address: 3CFH, Index 08H
Power-On Default: Undefined

Any bit programmed to 0 in this register will cause the corresponding bit in each of four memory planes to be immune to change. The data written into memory in this case is the data which was read in the previous cycle, and was stored in the processor latches. Any bit programmed to 1 allows unimpeded writes to the corresponding bits in the plane.

7	6	5	4	3	2	1	0
BIT MASK							

Bits 7–0 BIT MASK
A logical 0 means the corresponding bit of each plane in memory is set to the corresponding bit in the processor latches. A logical 1 means the corresponding bit of each plane in memory is set as specified by other conditions.

16.5 ATTRIBUTE CONTROLLER REGISTERS

The attribute controller registers are located at the same byte I/O address for writing address and data. An internal address flip-flop (AFF) controls the selection of either the attribute index or data registers. To initialize the address flip-flop (AFF), an I/O read is issued at address 3BAH or 3DAH. This presets the address flip-flop to select the index register. After the index register has been loaded by an I/O write to address 3C0H, AFF toggles and the next I/O write loads the data register. Every I/O write to address 3C0H toggles this address flip-flop. However, it does not toggle for I/O reads at address 3C0H or 3C1H. The Attribute Controller Index register is read at 3C0H, and the Attribute Controller Data register is read at address 3C1H.

Attribute Controller Index Register (ATR_AD)

Read/Write Address: 3C0H
Power-On Default: Undefined

This register is loaded with a binary index value that determines which attribute controller register will be accessed. This value is referred to as the "Index Number" of the AR register (AR0–14).

7	6	5	4	3	2	1	0
R	R	ENB PLT	ATTRIBUTE ADDRESS				

Bits 4–0 ATTRIBUTE ADDRESS
A binary value that points to the attribute controller register where data is to be written.

Bit 5 ENB PLT - Enable Video Display
0 = Video display access to the palette registers disabled. The Attribute Controller register can be accessed by the CPU
1 = Display video using the palette registers enabled (normal display operation). The palette registers (AR0–ARF) cannot be accessed by the CPU

This bit is effective only in 8-bit PA mode (CR67_4 = 0).

Bits 7–6 Reserved

Attribute Controller Data Register (ATR_DATA)

Read/Write Address: R: 3C1H/W: 3COH
Power-On Default: Undefined

This register is the data port for the attribute controller register indexed by the Attribute Controller Index register.

7	6	5	4	3	2	1	0
ATTRIBUTE DATA							

Bits 7–0 ATTRIBUTE DATA
Data to the attribute controller register indexed by the attribute controller address.

Palette Registers (PLT_REG) (AR00-0F)

Read/Write Address: 3C1H/3C0H, Index 00H-0FH
Power-On Default: Undefined

These are 16, 6-bit registers pointed to by the index and color code. They allow a dynamic mapping between the text attribute or graphics color input and the display color on the CRT screen.

7	6	5	4	3	2	1	0
		SECONDARY			PRIMARY		
= 0	= 0	SR	SG	SB	R	G	B

Bits 5–0 PALETTE COLOR
The six bit display color, bits 5-0 are output as SR, SG/I, SB/V, R, G and B, respectively.

Bits 7–6 Reserved = 0

Attribute Mode Control Register (ATR_MODE) (AR10)

Read/Write
Power-On Default: 00H

Address: 3C1H/3C0H, Index 10H

The contents of this register controls the attribute mode of the display function.

7	6	5	4	3	2	1	0
SEL V54	256 CLR	TOP PAN	= 0	ENB BLNK	ENB LGC	MONO ATRB	TX /GR

Bit 0 $\overline{TX}$/GR - Select Graphics Mode
0 = Selects text attribute control mode
1 = Selects graphics control mode

Bit 1 MONO ATRB - Select Monochrome Attributes
0 = Selects color display text attributes
1 = Selects monochrome display text attributes

Bit 2 ENB LGC - Enable Line Graphics
0 = The ninth dot of a text character (bit 0 of SR1 = 0) is the same as the background
1 = Special line graphics character codes enabled

When this bit is set to 1, it forces the ninth dot of a line graphics character to be identical to the eighth dot of the character. The line graphics character codes are C0H through DFH. For other characters, the ninth dot is the same as the background.

Bit 3 ENB BLNK - Enable Blinking
0 = Selects the background intensity for the text attribute input
1 = Selects blink attribute in text modes

This bit must also be set to 1 for blinking graphics modes. The blinking counter is operated by the vertical retrace counter (VRTC) input. It divides the VRTC input by 32. The blinking rates are ON for 16 VRTC clocks and OFF for 16 VRTC clocks. In the graphics mode, when blink is activated, the most significant color bit (bit 3) for each dot is inverted alternately, thus allowing two different colors to be displayed for 16 VRTC clocks each.

When the cursor is displayed in the text mode, it is blinked at a rate of ON for 8 VRTC clocks and OFF for 8 VRTC clocks (period by 16 frames). The displayed characters are independently blinked at the rate of 32 frames as above.

Bit 4 Reserved = 0

Bit 5 TOP PAN - Top Panning Enable
0 = Line compare has no effect on the output of the pixel panning register
1 = Forces the output of the pixel panning register to 0 after matching line compare until VSYNC occurs in the CRT controller. At the top of screen the output of the Pixel Panning register returns to its programmed value. This bit allows a top portion of a split screen to be panned.

Bit 6 256 CLR - Select 256 Color Mode
0 = 4 bits of video (translated to 6 bits by the palette) are output every internal dot-clock cycle
1 = Two 4-bit sets of video data are assembled to generate 8-bit video data at half the frequency of the internal dot-clock

Bit 7 SEL V54 - Select V[5:4]
0 = In VGA, mode, bits 5-4 of video output are generated by the attribute palette registers. Bits 7-6 of video output are always generated by bits 3-2 of AR14
1 = Bits 5-4 of video output are generated by bits 1-0 of AR14

Border Color Register (BDR_CLR) (AR11)

Read/Write Address: 3C1H/3C0H, Index 11H
Power-On Default: 00H

7	6	5	4	3	2	1	0
BORDER COLOR							

Bits 7–0 Border Color. This 8-bit register determines the border color displayed on the CRT screen. The border is an area around the screen display area.

This register is only effective in 8-bit PA modes (CR67_4 = 0). See also CR33_5.

Color Plane Enable Register (DISP_PLN) (AR12)

Read/Write Address: 3C1H/3C0H, Index 12H
Power-On Default: 00H

This register enables the respective video memory color plane 3–0 and selects video color outputs to be read back in the display status.

7	6	5	4	3	2	1	0
= 0	= 0	VDT-SEL 1	0	DISPLAY PLANE ENBL			

Bits 3–0 DISPLAY PLANE ENBL
A 0 in any of these bits forces the corresponding color plane bit to 0 before accessing the internal palette. A 1 in any of these bits enables the data on the corresponding color plane.

Bits 5–4 VDT-SEL - Video Test Select
These bits select two of the eight bit color outputs to be available in the Input Status 1 register. The output color combinations available on the status bits are as follows:

D STS MUX		STS 1	
Bit 5	**Bit 4**	**Bit 5**	**Bit 4**
0	0	Video 2	Video 0
0	1	Video 5	Video 4
1	0	Video 3	Video 1
1	1	Video 7	Video 6

Bits 7–6 Reserved = 0

Horizontal Pixel Panning Register (H_PX_PAN) (AR13)

Read/Write Address: 3C1H/3C0H, Index 13H
Power-On Default: 00H

This register specifies the number of pixels to shift the display data horizontally to the left. Pixel panning is available in both text and graphics modes. It is not available with Enhanced mode memory mappings (CR31_3 = 1).

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	NUMBER OF PAN SHIFT			

Bits 3–0 NUMBER OF PAN SHIFT
This register selects the number of pixels to shift the display data horizontally to the left. In the 9 pixels/character text mode, the output can be shifted a maximum shift of 8 pixels. In the 8 pixels/character text mode and all graphics modes, except 256 color mode, a maximum shift of 7 pixels is possible. In the 256 color mode, bit 0 of this register must be 0 resulting in only 4 panning positions per display byte. The panning is controlled as follows:

	Number of pixels shifted in		
Bits 3–0	**9 pixel/char.**	**8 pixel/char.**	**256 color mode**
0000	1	0	0
0001	2	1	–
0010	3	2	1
0011	4	3	–
0100	5	4	2
0101	6	5	–
0110	7	6	3
0111	8	7	–
1000	0	–	–

Bits 7–4 Reserved = 0

Pixel Padding Register (PX_PADD) (AR14)

Read/Write
Power-On Default: 00H

Address: 3C1H/3C0H, Index 14H

This register specifies the high-order bits of video output when pixel padding is enabled and disabled in the 256 color mode.

7	6	5	4	3	2	1	0
				PIXEL PADDING			
= 0	= 0	= 0	= 0	V7	V6	V5	V4

Bits 1–0 PIXEL PADDING V5, V4
These bits are enabled with a logical 1 of bit 7 of AR10, and can be used in place of the V5 and V4 bits from the Palette registers to form the 8-bit digital color value output.

Bits 3–2 PIXEL PADDING V7, V6
In all modes except 256 color mode, these bits are the two high-order bits of the 8-bit digital color value output.

Bits 7–4 Reserved = 0

16.6 RAMDAC REGISTERS

If the internal RAMDAC is used (CR55_3 = 0), all of the RAMDAC registers described in this section are physically located inside ViRGE. If the external RAMDAC is used (CR55_3 = 1), only the DAC Status Register (3C7H, Read Only) is physically located inside ViRGE/VX. The others are located in the RAMDAC. ViRGE/VX decodes these addresses for RAMDAC data byte steering.

DAC Mask Register (DAC_AD_MK)

Read/Write Address: 3C6H
Power-On Default: Undefined

This register is the pixel read mask register to select pixel video output. The CPU can access this register at any time. When using an external RAMDAC, an access to this register causes 10b to driven to the RS[1:0] outputs to the RAMDAC.

7	6	5	4	3	2	1	0
DAC ADDRESS MASK							

Bits 7–0 DAC ADDRESS MASK
The contents of this register are bit-wise logically ANDed with the pixel select video output (PA[7:0]). This register is initialized to FFH by the BIOS during a video mode set.

DAC Read Index Register (DAC_RD_AD)

Write Only Address: 3C7H
Power-On Default: Undefined

This register contains the pointer to one of 256 palette data registers and is used when reading the color palette. When using an external RAMDAC, an access to this register causes 11b to driven to the RS[1:0] outputs to the RAMDAC.

7	6	5	4	3	2	1	0
DAC READ ADDRESS							

Bits 7–0 DAC READ ADDRESS
Each time the color code is written to this register, it identifies that a read sequence will occur. A read sequence consists of three successive byte reads from the RAMDAC data register at I/O address 3C9H. The least significant 6 bits of each byte taken from the RAMDAC data register contain the corresponding color value, and the most significant 2 bits contain zeros. The order is red byte first, then green, and finally blue. The sequence of events for a read cycle is:

1. Write the color code to this register (RAMDAC Read Index) at address 3C7H.
2. The contents of the location in the color look-up table pointed to by the color code are transferred to the RAMDAC data register at address 3C9H.
3. Three bytes are read back from the RAMDAC data register.
4. The contents of this register auto-increment by one.
5. Go to step 2.

The effects of writing to the RAMDAC data register during a three-byte read cycle or reading from the RAMDAC data register during a 3-byte write cycle (i.e., interrupting the sequence) are undefined and may change the look-up table contents.

DAC Status Register (DAC_STS)

Read Only Address: 3C7H
Power-On Default: Undefined

When using an external RAMDAC, an access to this register causes 11b to driven to the RS[1:0] outputs to the RAMDAC.

7	6	5	4	3	2	1	0
= 0	= 0	= 0	= 0	= 0	= 0	DAC-STS	

Bits 1–0 DAC-STS - RAMDAC Cycle Status
The last executing cycle was:
00 = Write Palette cycle
11 = Read Palette cycle

Reads from the RAMDAC Write Index at address 3C8H or the DAC status register at address 3C7H do not interfere with read or write cycles and may take place at any time.

Bits 7–2 Reserved = 0

DAC Write Index Register (DAC_WR_AD)

Read/Write Address: 3C8H
Power-On Default: Undefined

When using an external RAMDAC, an access to this register causes 00b to driven to the RS[1:0] outputs to the RAMDAC.

7	6	5	4	3	2	1	0
DAC WRITE ADDRESS/GIP READ DATA							

Bits 7–0 DAC WRITE ADDRESS/GIP READ DATA
This register contains the pointer to one of 256 palette data registers and is used during a palette load. Each time the color code is written to this register, it identifies that a write sequence will occur. A write sequence consists of three successive byte writes to the DAC data register at I/O address 3C9H. The least significant 6 bits of each byte are concatenated to form the value placed in the 18-bit data register. The order is red byte first, then green, and finally blue. Once the third byte has been written, the value in the data register is written to the location pointed to by the color code. The sequence of events for a write cycle is:

1. Write the color code to this register (DAC Write Index) at address 3C8H.
2. Three bytes are written to the DAC Data register at address 3C9H.
3. The contents of the DAC data register are transferred to the location in the color look-up table pointed to by the color code.
4. The DAC Write Index register auto-increments by 1.
5. Go to step 2.

If bit 2 of the Extended RAMDAC Control register (CR55) is set to 1 to enable the General I/O Port read function for a VL-Bus configuration, a read of 3C8H retrieves data from an external input buffer. The data is transmitted directly to SD[7:0] for a VL-Bus configuration.

RAMDAC Data Register (DAC_DATA)

Read/Write Address: 3C9H
Power-On Default: Undefined

This register is a data port to read or write the contents of the location in the color look-up table pointed to by the DAC Read Index or the DAC Write Index registers. When using an external RAMDAC, an access to this register causes 01b to driven to the RS[1:0] outputs to the RAMDAC.

7	6	5	4	3	2	1	0
DAC READ/WRITE DATA							

Bits 7–0 DAC READ/WRITE DATA
To prevent "snow flicker" on the screen, an application reading data from or writing data to the DAC Data register should ensure that the $\overline{\text{BLANK}}$ input to the RAMDAC is asserted. This can be accomplished either by restricting data transfers to retrace intervals, checking bit 3 of the Input Status 1 register (3?AH) to determine when retrace is occurring, or by using the screen-off bit in the Clocking Mode register of the sequencer (bit 5 of SR1).

Section 17: Extended Sequencer Register Descriptions

The following registers are located in the Sequencer Register address space not used by the standard VGA.

In the following register descriptions, 'U' stands for undefined or unused and 'R' stands for reserved (write = 0, read = U).

See Appendix A for a table listing each register in this section and its page number.

Unlock Extended Sequencer Register (UNLK_EXSR) (SR8)

Read/Write Address: 3C5H, Index 08H
Power-On Default: 00H

Loading xxxx0110b (e.g., 06H) unlocks accessing of all the S3 extensions (SR9 - SR1C) to the standard VGA Sequencer register set. (x = don't care).

7	6	5	4	3	2	1	0
R	R	R	R	=0	=1	=1	=0

Extended Sequencer 9 Register (SR9)

Read/Write Address: 3C5H, Index 09H
Power-On Default: 00H

7	6	5	4	3	2	1	0
MMIO-ONLY	R	R	R	R	R	R	R

Bits 6-0 Reserved

Bit 7 MMIO-ONLY - Memory-mapped I/O register access only
0 = When MMIO is enabled, both programmed I/O and memory-mapped I/O register accesses are allowed
1 = When MMIO is enabled, only memory-mapped I/O register accesses are allowed to extended (non-standard VGA) registers. Both I/O and MMIO accesses can be made to standard VGA registers.

Extended Sequencer A Register (SRA)

Read/Write Address: 3C5H, Index 0AH
Power-On Default: 00H

7	6	5	4	3	2	1	0
2 MCLK	R	PD-NTRI	R	R	R	R	R

Bits 4-0 Reserved

Bit 5 PD-NTRI - PD[63:0] Not Tri-stated
0 = PD[63:0] tri-stated
1 = PD[63:0] not tri-stated

The default value of 0 reduces power consumption. The pins are enabled for output only as needed. Note that output pads for PD[63:29] also latch the most recent output state.

Bit 6 Reserved

Bit 7 2MCLK - 2 MCLK CPU writes to memory
0 = 3 MCLK memory writes
1 = 2 MCLK memory writes

Setting this bit to 1 improves performance for systems using an MCLK less than 57 MHz. For MCLK frequencies between 55 and 57 MHz, bit 7 of SR15 should also be set to 1 if linear addressing is being used.

Extended Sequencer B Register (SRB)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 0BH

7	6	5	4	3	2	1	0
ALT COLOR MODE				R	R	VAFC VCLKI	DOT= VCLKI

Bit 0 DOT = VCLKI - Dot clock = VCLKI
0 = Use internal dot clock
1 = Use VCLKI input for all internal dot clock functions

This bit is used for S3 test purposes only.

Bit 1 VAFC VCLKI - Use VCLKI input with VAFC
0 = Pixel data from pass-through feature connector latched by incoming VCLK
1 = Pixel data from VAFC latched by VCLKI input

Bits 3-2 Reserved

Bits 7-4 ALT COLOR MODE - Color Mode for feature connector input
0000 = 8-bit color, 1 pixel/VCLK
0011 = 15-bit color, 1 pixel/VCLK
0101 = 16-bit color, 1 pixel/VCLK

All other values are reserved.

Extended Sequencer D Register (SRD)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 0DH

This register provides feature connector control and also provides independent control of the HSYNC and VSYNC signals, therefore supporting the VESA DPMS (Display Power Management Control) standard.

7	6	5	4	3	2	1	0
VSY-CTL 1	0	HSY-CTL 1	0	R	R	R	EN-FEAT

Bit 0 EN-FEAT - Enable Feature Connector
0 = $\overline{\text{ENFEAT}}$ is high. VCLK, HSYNC and VSYNC are outputs.
1 = $\overline{\text{ENFEAT}}$ is low. The direction of VCLK is controlled by EVCLK and the direction of $\overline{\text{BLANK}}$, HSYNC and VSYNC is controlled by ESYNC. In both cases, assertion (low) specifies an input and a logic high specifies an output.

This bit is set to 1 to drive $\overline{\text{ENFEAT}}$ with a logic 0. This is used an a chip enable for the Scenic/MX2.

Bits 3–1 Reserved

Bits 5–4 HSY-CTL - HSYNC Control
00 = Normal operation
01 = HSYNC = 0
10 = HSYNC = 1
11 = Reserved

Bits 7–6 VSY-CTL - VSYNC Control
00 = Normal operation
01 = VSYNC = 0
10 = VSYNC = 1
11 = Reserved

MCLK Value Low Register (SR10)

Read/Write Address: 3C5H, Index 10H
Power-On Default: See description below.

The power-on default value for this register in conjunction with the power-on default value for SR11 generate an MCLK value of 45 MHz. All other MCLK values must be specified by programming of SR10 and SR11. Loading of a new value is enabled by either bit 0 or bit 5 of SR15.

7	6	5	4	3	2	1	0
R	PLL R VALUE		PLL N-DIVIDER VALUE				

Bits 4-0 PLL N-DIVIDER VALUE
These bits contain the binary equivalent of the integer (1-31) divider used to scale the input to the MCLK PLL. See Section 9 for a detailed explanation.

Bits 6-5 PLL R VALUE
These bits contain the binary equivalent of the integer (1, 2, 4, 8) range value used to scale the output of the MCLK PLL. See Section 9 for a detailed explanation.

Bit 7 Reserved

MCLK Value High Register (SR11)

Read/Write Address: 3C5H, Index 11H
Power-On Default: See description below.

The power-on default value for this register in conjunction with the power-on default value for SR10 generate an MCLK value of 45 MHz. All other MCLK values must be specified by programming of SR10 and SR11. Loading of a new value is enabled by either bit 0 or bit 5 of SR15.

7	6	5	4	3	2	1	0
R	PLL M-DIVIDER VALUE						

Bits 6-0 PLL M-DIVIDER VALUE
These bits contain the binary equivalent of the integer (1-127) divider used in the feedback loop of the MCLK PLL. See Section 9 for a detailed explanation.

Bit 7 Reserved

Enhanced Mode DCLK Value Low Register (SR12)

Read/Write Address: 3C5H, Index 12H
Power-On Default: Undefined

SR12 and SR13 are selected as the source of the DCLK PLL parameter values when 3C2H_3-2 = 11b. This setting is used for all Enhanced modes. Loading of new values occurs when either SR15_1 is set to 1 or SR15_5 is programmed to 1 and then 0.

7	6	5	4	3	2	1	0
PLL R VALUE			PLL N-DIVIDER VALUE				

Bits 4-0 PLL N-DIVIDER VALUE
These bits contain the binary equivalent of the integer divider used to scale the input to the DCLK PLL. See Section 8 for a detailed explanation.

Bits 7-5 PLL R VALUE
These bits contain the binary equivalent of the integer range value used to scale the output of the DCLK PLL. See Section 8 for a detailed explanation.

Enhanced Mode DCLK Value High Register (SR13)

Read/Write Address: 3C5H, Index 13H
Power-On Default: Undefined

SR12 and SR13 are selected as the source of the DCLK PLL parameter values when 3C2H_3-2 = 11b. This setting is used for all Enhanced modes. Loading of new values occurs when either SR15_1 is set to 1 or SR15_5 is programmed to 1 and then 0.

7	6	5	4	3	2	1	0
R	PLL M-DIVIDER VALUE						

Bits 6-0 PLL M- DIVIDER VALUE
These bits contain the binary equivalent of the integer divider used in the feedback loop of the DCLK PLL. See Section 8 for a detailed explanation.

Bit 7 Reserved

CLKSYN Control 1 Register (SR14)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 14H

7	6	5	4	3	2	1	0
EXT DCLK	EXT MCLK	PA16 SEL	CLR CNT	CLK TEST	EN CNT	MPLL PD	DPLL PD

Bit 0 DPLL PD - Power Down DCLK PLL
0 = DCLK PLL powered
1 = DCLK PLL powered down

This bit is used for S3 test purposes only.

Bit 1 MPLL PD - Power Down MCLK PLL
0 = MCLK PLL powered
1 = MCLK PLL powered down

This bit is used for S3 test purposes only.

Bit 2 EN CNT - Enable Clock Synthesizer Counters
0 = Clock synthesizer counters disabled
1 = Clock synthesizer counters enabled

See Section 9.4 for a description of the clock testing procedure.

Bit 3 CLK TEST - Clock Test
0 = Test DCLK
1 = Test MCLK

If DCLK is selected, ensure that 3C2H_3-2 = 11b.

Bit 4 CLR CNT - Clear Clock Synthesizer Counter
0 = No effect
1 = Clear the clock synthesizer counter

Bit 5 PA16 SEL - Pin A16 Function Select
0 = Pin A16 functions normally
1 = Pin A16 is tri-stated

Setting this bit to 1 allows pin A16 to act as an MCLK input. This is enabled by setting bit 6 of this register to 1.

Bit 6 EXT MCLK - External MCLK Select
0 = MCLK provided by internal PLL
1 = MCLK is input on pin A16

This bit can also be set to 1 at reset via power-on strapping of PD11. An external MCLK is only used for S3 test purposes.

Bit 7 EXT DCLK - External DCLK Select
0 = DCLK provided by internal PLL
1 = DCLK is input on pin C21.

This bit can also be set to 1 at reset via power-on strapping of PD11. An external DCLK is only used for S3 test purposes.

CLKSYN Control 2 Register (SR15)

Read/Write Address: 3C5H, Index 15H
Power-On Default: 00H

7	6	5	4	3	2	1	0
2 CYC MWR	DCLK\ INV	CLK LOAD	DCLK/ 2	VCLK OUT	MCLK OUT	DRFQ EN	MFRQ EN

Bit 0 MFRQ EN - Enable new MCLK frequency load
0 = Register bit clear
1 = Load new MCLK frequency

When new MCLK PLL values are programmed, this bit can be set to 1 to load these values in the PLL. The loading may be delayed a small but variable amount of time. This bit should be cleared to 0 after loading to prevent repeated loading. Alternately, use bit 5 of this register to produce an immediate load.

Bit 1 DFRQ EN - Enable new DCLK frequency load
0 = Register bit clear
1 = Load new DCLK frequency

When new DCLK PLL values are programmed, this bit can be set to 1 to load these values in the PLL. The source of the PLL values is specified by 3C2H_3-2. The loading may be delayed a small but variable amount of time. This bit should be programmed to 1 at power-up to allow loading of the VGA DCLK value and then left at this setting. Use bit 5 of this register to produce an immediate load.

Bit 2 MCLK OUT - Output internally generated MCLK
0 = Pin B8 functions normally
1 = Pin B8 outputs the internally generated MCLK

This is used only for testing.

Bit 3 VCLK OUT - VCLK direction determined by $\overline{\text{EVCLK}}$
0 = Pin D17 outputs the internally generated VCLK regardless of the state of $\overline{\text{EVCLK}}$
1 = VCLK direction is determined by the $\overline{\text{EVCLK}}$ signal

This bit is effective only when the LPB feature connector is enabled.

Bit 4 DCLK/2 - Divide DCLK by 2
0 = DCLK unchanged
1 = Divide DCLK by 2

Either this bit or bit 6 of this register must be set to 1 for clock doubled RAMDAC operation (mode 0001).

Bit 5 CLK LOAD - MCLK, DCLK load
0 = Clock loading is controlled by bits 0 and 1 of this register
1 = Load MCLK and DCLK PLL values immediately

To produce an immediate MCLK and DCLK load, program this bit to 1 and then to 0. The source of the PLL values is specified by 3C2H_3-2. This register must never be left set to 1.

Bit 6 DCLK INV - Invert DCLK
0 = DCLK unchanged
1 = Invert DCLK

Either this bit or bit 4 of this register must be set to 1 for clock doubled RAMDAC operation (mode 0001).

Bit 7 2 CYC MWR - Enable 2 cycle memory write
0 = 3 MCLK memory write
1 = 2 MCLK memory write

Setting this bit to 1 bypasses the VGA logic for linear addressing when bit 7 of SRA is set to 1. This can allow 2 MCLK operation for MCLK frequencies between 55 and 57 MHz.

CLKSYN Test High Register (SR16)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 16H

7	6	5	4	3	2	1	0
CLOCK TEST RESULTS HIGH BYTE							

Bits 7-0 CLOCK TEST RESULTS HIGH BYTE

See Section 9.4 for a description of how to use this register.

CLKSYN Test Low Register (SR17)

Read Only
Power-On Default: 00H

Address: 3C5H, Index 17H

7	6	5	4	3	2	1	0
CLOCK TEST RESULTS LOW BYTE							

Bits 7-0 CLOCK TEST RESULTS LOW BYTE

See Section 9.4 for a description of how to use this register.

RAMDAC/CLKSYN Control Register (SR18)

Read/Write
Power-On Default: 00H

Address: 3C5H, Index 18H

7	6	5	4	3	2	1	0
CLKx 2	LUT WR	DAC PD	TST BLUE	TST GRN	TST RED	TST RST	TST EN

Bit 0 TST EN - Enable Signature Test Counter
0 = RAMDAC signature test counter disabled
1 = RAMDAC signature test counter enabled

Bit 1 TST RST - Reset Signature Test Counter
0 = No effect
1 = Reset the RAMDAC signature test counter

Bit 2 TST RED - Test Red Data
0 = No effect
1 = Place red data on internal data bus

The red signature data is read via CR6E.

Bit 3 TST GRN - Test Green Data
0 = No effect
1 = Place green data on internal data bus

The red signature data is read via CR6E.

Bit 4 TST BLUE - Test Blue Data
0 = No effect
1 = Place blue data on internal data bus

The blue signature data is read via CR6E.

Bit 5 DAC PD - RAMDAC power-down
0 = RAMDAC powered
1 = RAMDAC powered-down

When the RAMDAC is powered down, the RAMDAC memory retains its data.

Bit 6 LUT WR - LUT write cycle control
0 = 2 DCLK LUT write cycle (default)
1 = 1 DCLK LUT write cycle

Bit 7 CLKx2 - Enable clock doubled mode
0 = RAMDAC clock doubled mode (0001) disabled
1 = RAMDAC clock doubled mode (0001) enabled

This bit must be set to 1 when mode 0001 is specified in bits 7-4 of CR67 or SRC. Either bit 4 or bit 6 of SR15 must also be set to 1.

RAM Test Register (SR19)

See Bit Definitions Address: 3C5H, Index 19H
Power-On Default: 00H

This register is used only for S3 testing.

7	6	5	4	3	2	1	0
HC FETCH DELAY			SH DELAY		R	RTD	RTST

Bit 0 RTST - RAM Test Enable (Read/Write)
0 = Disable RAM test
1 = Enable RAM test

Bit 1 RTD - RAM Test Done (Read Only)
0 = No meaning
1 = RAM test completed

Bit 2 Reserved

Bit 4-3 SH DELAY - Delay HSYNC During Streams Processor Operation
00 = 10 DCLKs delay
01 = 12 DCLKs delay
10 = 14 DCLKs delay
11 = 16 DCLKs delay

This delay is functional only when the Streams Processor is enabled and is in addition to the HSYNC delay specified via CR63_7-4.

Bits 7-5 HC FETCH DELAY - Hardware Cursor Fetch Delay
0000 = 9 character clocks delay
0001 = 8 character clocks delay
0010 = 7 character clocks delay
0011 = 6 character clocks delay
0100 = 5 character clocks delay
0101 = 4 character clocks delay
0110 = 3 character clocks delay
0111 = 2 character clocks delay
100x = 10 character clocks delay
101x = 1 character clock delay
110x = 0 character clocks delay
111x = -1 character clock delay

Extended Sequencer 1A Register (SR1A)

Read/Write Address: 3C5H, Index 1AH
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	BP	DDPS	PLL VCO ADJUST			

Bits 3-0 PLL VCO ADJUST

These bits allow adjustment of the maximum PLL VCO frequency. S3 will provide the recommended values.

Bit 4 DDPS - Disable DAC Power Saving
0 = Enable DAC power saving
1 = Disable DAC power saving

The default (0) setting allows the DACs to be automatically powered down during the blanking period to reduce power consumption.

Bit 5 BP - $\overline{\text{BLANK}}$ Pedestal
0 = No $\overline{\text{BLANK}}$ pedestal
1 = Add $\overline{\text{BLANK}}$ Pedestal to RAMDAC output

Bits 7-6 Reserved

Extended Sequencer 1C Register (SR1C)

Read/Write Address: 3C5H, Index 1CH
Power-On Default: 00H

The bits in this register are effective only in LPB mode.

7	6	5	4	3	2	1	0
R	R	R	R	R	R	SIGSEL	

Bits 1–0 SIGSEL - Signal Select
00 = Pin A16 is $\overline{\text{ENFEAT}}$; pin A17 is $\overline{\text{ROMEN}}$, pin B8 is $\overline{\text{STWR}}$
01 = Pin A16 is reserved; A17 is $\overline{\text{ROMEN}}$, pin B8 is $\overline{\text{STWR}}$
10 = Pin A16 is GOP0; pin A17 is $\overline{\text{ROMEN}}$, pin B8 is GOP1
11 = Pin A16 is GOP0; pin A17 is $\overline{\text{ROMEN}}$, pin B8 is GOP1

GOP0 and GOP1 are bits 0-1 of the General Output Port register (CR5C).

When the system powers up with a default value of 00b for bits 1-0, pins A16 and B8 will be driven high.

Bits 7–2 Reserved

VGA DCLK0 Value Low Register (SR22)

Read/Write Address: 3C5H, Index 22H
Power-On Default: See description below

SR22 and SR23 are selected as the source of the DCLK PLL parameter values when 3C2H_3-2 = 00b. The power-on default value for this register in conjunction with the power-on default value for SR23 generate a DCLK value of 25.175 MHz. Loading of new values occurs when either SR15_1 is set to 1 or SR15_5 is programmed to 1 and then 0.

7	6	5	4	3	2	1	0
PLL R VALUE			PLL N-DIVIDER VALUE				

Bits 4-0 N-DIVIDER VALUE
These bits contain the binary equivalent of the integer (1-31) divider used to scale the input to the DCLK PLL. See Section 8 for a detailed explanation.

Bits 7-5 PLL R VALUE
These bits contain the binary equivalent of the integer (1, 2, 4, 8) range value used to scale the output of the DCLK PLL. See Section 8 for a detailed explanation.

VGA DCLK0 Value High Register (SR23)

Read/Write Address: 3C5H, Index 23H
Power-On Default: See description below.

SR22 and SR23 are selected as the source of the DCLK PLL parameter values when 3C2H_3-2 = 00b. The power-on default value for this register in conjunction with the power-on default value for SR22 generate a DCLK value of 25.175 MHz. Loading of new values occurs when either SR15_1 is set to 1 or SR15_5 is programmed to 1 and then 0.

7	6	5	4	3	2	1	0
R	PLL M-DIVIDER VALUE						

Bits 6-0 PLL M- DIVIDER VALUE
These bits contain the binary equivalent of the integer (1-127) divider used in the feedback loop of the DCLK PLL. See Section 8 for a detailed explanation.

Bit 7 Reserved

VGA DCLK0 Value Low Register (SR24)

Read/Write Address: 3C5H, Index 24H
Power-On Default: See description below

SR24 and SR25 are selected as the source of the DCLK PLL parameter values when 3C2H_3-2 = 01b. The power-on default value for this register in conjunction with the power-on default value for SR25 generate a DCLK value of 28.322 MHz. Loading of new values occurs when either SR15_1 is set to 1 or SR15_5 is programmed to 1 and then 0.

7	6	5	4	3	2	1	0
PLL R VALUE			PLL N-DIVIDER VALUE				

Bits 4-0 N-DIVIDER VALUE
These bits contain the binary equivalent of the integer divider used to scale the input to the DCLK PLL. See Section 8 for a detailed explanation.

Bits 7-5 PLL R VALUE
These bits contain the binary equivalent of the integer range value used to scale the output of the DCLK PLL. See Section 8 for a detailed explanation.

VGA DCLK0 Value High Register (SR25)

Read/Write Address: 3C5H, Index 25H
Power-On Default: See description below.

SR24 and SR25 are selected as the source of the DCLK PLL parameter values when 3C2H_3-2 = 01b. The power-on default value for this register in conjunction with the power-on default value for SR24 generate a DCLK value of 28.322 MHz. Loading of new values occurs when either SR15_1 is set to 1 or SR15_5 is programmed to 1 and then 0.

7	6	5	4	3	2	1	0
R	PLL M-DIVIDER VALUE						

Bits 6-0 PLL M- DIVIDER VALUE
These bits contain the binary equivalent of the integer divider used in the feedback loop of the DCLK PLL. See Section 8 for a detailed explanation.

Bit 7 Reserved

Section 18: Extended CRTC Register Descriptions

These registers are located in CRT Controller address space at locations not used by the IBM® VGA. All of these registers are read/write protected at power-up by hardware reset. In order to read/write these registers, CR38 and/or CR39 must be loaded with a changed key pattern (see the register description). The registers will remain unlocked until the key pattern is reset by altering a significant bit.

In the following register descriptions, 'R' stands for reserved (write =0, read = undefined). See Appendix A for a table listing each register in this section and its page number.

Device ID High Register (CR2D)

Read Only Address: 3?5H, Index 2DH
Power-On Default: 88H

This register should contain the same value as the upper byte of the PCI Device ID (Index 02H) register.

7	6	5	4	3	2	1	0
CHIP ID HIGH (88H)							

Bits 7–0 CHIP ID HIGH

Device ID Low Register (CR2E)

Read Only Address: 3?5H, Index 2EH
Power-On Default: 3DH

7	6	5	4	3	2	1	0
CHIP ID LOW (3DH)							

Bits 7–0 CHIP ID LOW

Revision Register (CR2F)

Read Only Address: 3?5H, Index 2FH
Power-On Default: See Description

7	6	5	4	3	2	1	0
REVISION LEVEL							

Bits 7–0 REVISION LEVEL

Hardwired to 00H for the first version on ViRGE/VX. This will change with later steppings.

Chip ID/REV Register (CHIP-ID/REV) (CR30)

Read Only Address: 3?5H, Index 30H
Power-On Default: E1H

7	6	5	4	3	2	1	0
CHIP ID				REVISION STATUS			

Bits 7–0 CHIP ID AND REVISION STATUS

Memory Configuration Register (MEM_CNFG) (CR31)

Read/Write Address: 3?5H, Index 31H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	HST DFF	OLD-DSAD 17	16	ENH MAP	VGA 16B	R	CPUA BASE

Bit 0 CPUA BASE - Enable Base Address Offset
0 = Address offset bits 3-0 of CR35 and bits 3-2 of CR51 or the new address offset bits (5-0 of CR6A) are disabled
1 = Address offset bits 3-0 CR35 and bits 3-2 of CR51 or the new address offset bits (5-0 of CR6A) are enabled for specifying the 64K page of display memory. Bits 5-0 of CR6A are used if this field contains a non-zero value. This allows access to up to 4 MBytes of display memory through a 64K window.

Bit 1 Reserved

Bit 2 VGA 16B - Enable VGA 16-bit Memory Bus Width
0 = 8-bit memory bus operation
1 = Enable 16-bit bus VGA memory read/writes

This is useful in VGA text modes when VGA graphics controller functions are typically not used.

Bit 3 ENH MAP - Use Enhanced Mode Memory Mapping
0 = Force IBM VGA mapping for memory accesses
1 = Force Enhanced Mode mappings

Setting this bit to 1 overrides the settings of bit 6 of CR14 and bit 3 of CR17 and causes the use of doubleword memory addressing mode. Also, the function of bits 3-2 of GR6 is overridden with a fixed 64K map at A0000H.

Bits 5–4 OLD-DSAD 17, 16 - Old Display Start Address Bits 17-16
Bits 17-16 of start address (CRC, CRD) and cursor location (CRE, CRF)

Bits 1-0 of the Extended System Control 2 register (CR51) are bits 19-18 of the address and enable access to up to 4 MBytes of display memory. If a value is programmed into bits 3-0 of the Extended System Control 3 register (CR69), this value becomes the upper 4 bits of the display start base address and bits 5-4 of CR31 and bits 1-0 of CR51 are ignored.

Bit 6 HST DFF - Enable High Speed Text Display Font Fetch Mode
0 = Normal font access mode
1 = Enable high speed text display

Setting this bit to 1 is only required for DCLK rates greater than 40 MHz. See bit 5 of CR3A.

Bit 7 Reserved

Backward Compatibility 1 Register (BKWD_1) (CR32)

Read/Write Address: 3?5H, Index 32H
Power-On Default: 00H

7	6	5	4	3	2	1	0
SCTS	VGA FXPG	R	INT EN	R	R	R	R

Bits 3–0 Reserved

Bit 4 INT EN -Interrupt Enable
0 = All interrupt generation disabled
1 = Interrupt generation enabled

Bit 5 Reserved

Bit 6 VGA FXPG - Use Standard VGA Memory Wrapping
0 = Memory accesses extending past a 256K boundary do not wrap
1 = Memory accesses extending past a 256K boundary wrap at the boundary

The standard 256K VGA memory page always ends on a natural 256K boundary and accesses beyond this boundary will wrap. If the starting address is moved via bits 4-0 of CR69 (or bits 5-4 of CR31 and bits 1-0 of CR51), the 256K page may not end on a 256K boundary and accesses past the boundary will not wrap. This is the case when this bit is cleared to 0. For standard VGA compatibility when the page base address is moved, this bit is set to 1 to cause wrapping at a 256K boundary.

Bit 7 SCTS - Serial Control Tri-State
0 = Normal operation
1 = All SC, $\overline{SE}$ and DSF pins are tri-stated.

Backward Compatibility 2 Register (BKWD_2) (CR33)

Read/Write Address: 3?5H, Index 33H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	LOCK PLTW	BDR SEL	LOCK DACW	VCLK= -DCK	R	DIS VDE	R

Bit 0 Reserved

Bit 1 DIS VDE - Disable Vertical Display End Extension Bits Write Protection
0 = VDE protection enabled
1 = Disables the write protect setting of the bit 7 of CR11 on bits 1 and 6 of CR7

Bit 2 Reserved

Bit 3 VCLK = -DCK - VCLK is Inverted DCLK
0 = VCLK is the external VCLK (pass-through feature connector clock input enabled) or is divided by 2 for 4 bits/pixel modes (see bit 6 of AR10 or bit 4 of CR3A) or is the internal DCLK (if neither of the first two cases apply)
1 = VCLK is forced to inverted DCLK

Bit 4 LOCK DACW - Lock RAMDAC Writes
0 = Enable writes to RAMDAC registers
1 = Disable writes to RAMDAC registers

Bit 5 BDR SEL - Blank/Border Select
0 = $\overline{BLANK}$ active time is defined by CR2 and CR3
1 = $\overline{BLANK}$ is active during entire display inactive period (no border)

Bit 6 LOCK PLTW - Lock Palette/Border Color Registers
0 = Unlock Palette/Border Color registers
1 = Lock Palette/Border Color registers

Bit 7 Reserved

Backward Compatibility 3 Register (BKWD_3) (CR34)

Read/Write Address: 3?5H, Index 34H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	R	ENB DTPC	R	PCI RET	PCI ABT	PCI SNP

Bit 0 PCI SNP - PCI DAC snoop method
0 = Handling of PCI master aborts and retries during DAC cycles controlled by bits 1 and 2 of this register
1 = PCI master aborts and retries are not handled during DAC cycles

Bit 1 PCI ABT - PCI master aborts during DAC cycles
0 = PCI master aborts handled during DAC cycles
1 = PCI master aborts not handled during DAC cycles

Bit 0 of this register must be cleared to 0 for this bit to be effective.

Bit 2 PCI RET - PCI retries during DAC cycles
0 = PCI retries handled during DAC cycles
1 = PCI retries not handled during DAC cycles

Bit 0 of this register must be cleared to 0 for this bit to be effective.

Bit 3 Reserved

Bit 4 ENB SFF - Enable Data Transfer Position Control
0 = Data transfer position defined by CR0 and CR4 programming
1 = Data transfer position defined by CR3B

This bit selects what register defines the timing of pixel data transfers from the DRAM side of the VRAM to the serial output side.

Bits 7–5 Reserved

CRT Register Lock Register (CRTR_LOCK) (CR35)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 35H

7	6	5	4	3	2	1	0
		LOCK	LOCK	OLD-CPU-BASE-ADDRESS			
R	R	HTMG	VTMG	17	16	15	14

Bits 3–0 OLD-CPU-BASE-ADDRESS
CPU Base Address bits 17-14. These four bits define the CPU address base in 64 KByte units of display memory. These bits are added with CPU address bit 17 (MSB of video memory addressing) to bit 14 for display buffer accesses.

Bits 3-2 of the Extended System Control 2 register (CR51) are bits 19-18 of the address and enable access to up to 4 MBytes of display memory. If a value is programmed into bits 5-0 of the Extended System Control 4 register (CR6A), this value becomes the upper 6 bits of the CPU base address and bits 3-0 of CR35 and bits 3-2 of CR51 are ignored.

Bit 4 LOCK VTMG - Lock Vertical Timing Registers
0 = Vertical timing registers are unlocked
1 = The following vertical timing registers are locked:

CR6
CR7 (bits 7,5,3,2,0)
CR9 (bit 5)
CR10
CR11 (bits 3-0)
CR15
CR16

CR6, CR7 registers are also locked by bit 7 of the Vertical Retrace End register (CR11).

Bit 5 LOCK HTMG - Lock Horizontal Timing Registers
0 = Horizontal timing registers are unlocked
1 = The following horizontal timing registers are locked:

CR00
CR1
CR2
CR3
CR4
CR5
CR17 (bit 2)

All these registers (except bit 2 of CR17) are also locked by bit 7 of the Vertical Retrace End register (CR11).

Bit 7-6 Reserved

Configuration 1 Register (CONFG_REG1) (CR36)

Read/Write* Address: 3?5H, Index 36H
Power-On Default: Depends on Strapping

* Bits 1-0 are read only. The other bits can be written only after 0A5H is written to CR39.

This register samples the reset state from PD bus pins [7:0]. Other configuration strapping bits are found in CR37, CR68 and CR6F.

7	6	5	4	3	2	1	0
8B	MEM SIZE		R	MEM MODE		R	R

Bits 1- 0 Reserved

Bit 3-2 MEM MODE - Memory Page Mode Select
00 = VRAM 1-cycle EDO mode
01 = Reserved
10 = VRAM 2-cycle EDO mode
11 = VRAM Fast page mode

Bit 4 Reserved

Bits 6-5 MEM Size - Memory Size
00 = 2 MBytes
01 = 4 MBytes
10 = 6 MBytes
11 = 8 MBytes

Bit 7 8-C - 8-Column Block Write Support
0 = VRAM does not support 8-column block writes
1 = VRAM supports 8-column block writes

If this bit is cleared to 0, block writes cannot be enabled.

Configuration 2 Register (CONFG_REG2) (CR37)

Read/Write* Address: 3?5H, Index 37H
Power-On Default: Depends on Strapping

* These bits can be written only after 0A5H is written to CR39.

This register samples the reset state from PD bus pins [15:8]. Other configuration strapping bits are found in CR36, CR68 and CR6F.

7	6	5	4	3	2	1	0
R	DRAM		R	CS	VBS	TEST	R

Bit 0 Reserved

Bit 1 TEST - Test Mode
0 = Test mode enabled
1 = Normal operation

Enabling test mode tri-states all outputs and sets all bidirectional pins to inputs.

Bit 2 Reserved

Bit 3 CS - Clock Select
0 = Use external DCLK on PDOWN (C21) and external MCLK on ENFEAT (A16) (test purposes only)
1 = Use internal DCLK, MCLK

Bit 4 Reserved

Bits 6-5 DRAM
00 = Reserved
01 = 4 MBytes DRAM
10 = 2 MBytes DRAM
11 = 0 MBytes DRAM

This specifies the amount of DRAM in a mixed VRAM/DRAM configuration. The value is subtracted from the total memory size to determine the amount of VRAM present.

Bit 7 Reserved

Register Lock 1 Register (REG_LOCK1) (CR38)

Read/Write Address: 3?5H, Index 38
Power-On Default: 00H

Loading 01xx10xx (e.g., 48H) into this register unlocks the extended CRTC registers from CR2D through CR3F for read/writes. (x = don't care)

7	6	5	4	3	2	1	0
= 0	= 1			= 1	= 0		

Register Lock 2 Register (REG_LOCK2) (CR39)

Read/Write Address: 3?5H, Index 39
Power-On Default: 00H

Loading 101xxxxx (e.g., A0H) unlocks the extended CRTC registers from CR40 throught CRFF for reading/writing (x = don't care). Loading A5H allows bits 7-2 of CR36, bits 7-0 of CR37 and bits 7-0 of CR68 to be written.

7	6	5	4	3	2	1	0
= 1	= 0	= 1					

Miscellaneous 1 Register (MISC_1) (CR3A)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 3AH

7	6	5	4	3	2	1	0
PCIRB DISA	R	HST DFW	ENH 256	TOP MEM	ENB RFC	REF-CNT 1	REF-CNT 0

Bits 1–0 REF-CNT - Alternate Refresh Count Control
00 = Refresh Count 0
01 = Refresh Count 1
10 = Refresh Count 2
11 = Refresh Count 3

If enabled by setting bit 2 of this register to 1, these bits override the refresh count in bit 6 of CR11 and specify the number of refresh cycles per horizontal line.

Bit 2 ENB RFC - Enable Alternate Refresh Count Control
0 = Alternate refresh count control (bits 1-0) is disabled
1 = Alternate refresh count control (bits 1-0) is enabled

Bit 3 TOP MEM - Enable Top of Memory Access
0 = Top of memory access disabled
1 = Simultaneous VGA text and Enhanced modes are enabled. CPU and CRTC accesses are then directed to the top 32- or 64-KByte area of display memory depending on whether address bit 13 is 0 or 1 respectively.

Bit 4 ENH 256 - Enable 8 Bits/Pixel or Greater Color Enhanced Mode
0 = Attribute controller shift registers configured for 4-bit modes
1 = Attribute controller shift register configured for 8-, 16- and 24-bit color Enhanced modes

Bit 5 HST DFW - Enable High Speed Text Font Writing
0 = Disable high speed text font writing
1 = Enable high speed text font writing

Setting this bit to 1 is only required for DCLK rates greater than 40 MHz. See bit 6 of CR31.

Bit 6 Reserved

Bit 7 PCIRB DISA - PCI Read Bursts Disabled
0 = PCI read burst cycles enabled
1 = PCI read burst cycles disabled

Note: Bit 7 of CR66 must be set to 1 before this bit is set to 1.

Data Transfer Execute Position Register (DT_EX_POS) (CR3B)

Read/Write Address: 3?5H, Index 3BH
Power-On Default: 00H

The Data Transfer Execute Position is a 9-bit value that specifies horizontal position in character clocks of the start of the VRAM read data transfer cycle. Bit 6 of CR5D is bit 8 of this field. This register is enabled by bit 4 of CR34.

7	6	5	4	3	2	1	0
DATA TRANSFER EXECUTE POSITION							

Bits 7–0 DATA TRANSFER EXECUTE POSITION.
9-bit Value = the horizontal position in character clocks of the start of the VRAM read data transfer cycle. This register contains the least significant 8 bits of this value.

Interlace Retrace Start Register (IL_RTSTART) (CR3C)

Read/Write Address: 3?5H, Index 3CH
Power-On Default: 00H

This value allows determination of the even/odd row active display starting positions when operating in an interlaced mode. This register is enabled by bit 5 of CR42.

7	6	5	4	3	2	1	0
INTERLACE RETRACE START POSITION							

Bits 7–0 INTERLACE RETRACE START POSITION
Value = offset in terms of character clocks for Interlaced mode start/end in even/odd frames.

System Configuration Register (SYS_CNFG) (CR40)

Read/Write
Power-On Default: 30H

Address: 3?5H, Index 40H

7	6	5	4	3	2	1	0
=0	=0	R = 1	R	R	R	R	EN ENH

Bit 0 EN ENH - Enable Enhanced Register Access
0 = Enhanced register access disabled
1 = Enhanced register access enabled

Bits 4–1 Reserved

Bit 5 Reserved = 1 (Default)

Bits 7–6 Reserved = 00b

BIOS Flag Register (BIOS_FLAG) (CR41)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 41H

7	6	5	4	3	2	1	0
BIOS-FLAG-REGISTER-1							

Bits 7–0 BIOS-FLAG-REGISTER-1
Used by the video BIOS.

Mode Control Register (MODE_CTL) (CR42)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 42H

7	6	5	4	3	2	1	0
R	R	INTL MODE	R	R	R	R	R

Bits 4–0 Reserved

Bit 5 INTL MODE - Interlaced Mode
0 = Noninterlaced
1 = Interlaced

This bit enables the function of CR3C.

Bits 7–6 Reserved

Extended Mode Register (EXT_MODE) (CR43)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 43H

7	6	5	4	3	2	1	0
HCTR X2	R	R	R	R	OLD LSW8	R	R

Bits 1–0 Reserved

Bit 2 OLD LSW8 - Logical Screen Width Bit 8
This is an extension of the Offset (Screen Width) register (CR13). This is disabled if bits 5-4 of the Extended System Control 2 register (CR51) are not 00b.

Bits 6–3 Reserved

Bit 7 HCTR X2 - Horizontal Counter Double Mode
0 = Disable horizontal counter double mode
1 = Enable horizontal counter double mode (horizontal CRT parameters are doubled)

Hardware Graphics Cursor Mode Register (HGC_MODE) (CR45)

Read/Write Address: 3?5H, Index 45H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	R	HWGC 1280	R	R	CDE	HWGC ENB

Bit 0 HWGC ENB - Hardware Graphics Cursor Enable
0 = Hardware graphics cursor disabled in any mode
1 = Hardware graphics cursor enabled in Enhanced mode

If an external RAMDAC is used, the RAMDAC must also provide the hardware cursor.

Bit 1 CDE - Cursor Delay Extension

This is the high order bit 3 of the hardware cursor fetch delay field. Bits 2-0 are in SR19_7-5.

Bits 3-2 Reserved

Bit 4 HWGC 1280 - Hardware Cursor Right Storage
0 = Function disabled
1 = For 4 bits/pixel, the last 256 bytes in each 1-KByte line of the hardware cursor start address become the hardware graphics cursor storage area. For 8 bits/pixel, the last 512 bytes in each 2-KByte line of the hardware cursor start address become the hardware graphics cursor storage area. In either case, bits 1-0 of CR4D must be 11b.

Bits 7–5 Reserved

Hardware Graphics Cursor Origin-X Registers (HWGC_ ORGX(H)(L)) (CR46, CR47)

Read/Write Address: 3?5H, Index 46H, 47H
Power-On Default: 0000H

The high order three bits are written into CR46 and the low order byte is written into CR47.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	HWGC ORG X (H)			HWGC ORG X (L)							

Bits 10–0 HWGC ORG X(H) (L) - X-Coordinate of Cursor Left Side

Bits 15–11 Reserved

Hardware Graphics Cursor Origin-Y Registers (HWGC_ORGY(H)(L)) (CR48, CR49)

Read/Write Address: 3?5H, Index 48H, 49H
Power-On Default: Undefined

The high order three bits are written into CR48 and the low order byte is written into CR49.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	HWGC ORG Y (H)			HWGC ORG Y (L)							

Bits 10–0 HWGC ORG Y (H)(L) - Y-Coordinate of Cursor Upper Line
The cursor X, Y position is registered upon writing HWGC ORG Y (H).

Bits 15–11 Reserved

Hardware Graphics Cursor Foreground Color Stack Register (HWGC_FGSTK) (CR4A)

Read/Write Address: 3?5H, Index 4AH
Power-On Default: Undefined

7	6	5	4	3	2	1	0
TRUE COLOR FOREGROUND STACK (0-2)							

Bits 7–0 TRUE COLOR FOREGROUND STACK (0-2)

Three foreground color registers are stacked at this address. The stack pointer (common with CR4B) is reset to 0 by reading the Hardware Graphics Cursor Mode register (CR45). Each write to this register (CR4A) increments the stack pointer by 1, so three writes provide 24 bits of true color information.

Hardware Graphics Cursor Background Color Stack Register (HWGC_BGSTK) (CR4B)

Read/Write Address: 3?5H, Index 4BH
Power-On Default: Undefined

7	6	5	4	3	2	1	0
TRUE COLOR BACKGROUND STACK (0-2)							

Bits 7–0 TRUE COLOR BACKGROUND STACK (0-2)

Three background color registers are stacked at this address. The stack pointer (common with CR4A) is reset to 0 by reading the Hardware Graphics Cursor Mode register (CR45). Each write to this register (CR4B) increments the stack pointer by 1, so three writes provide 24 bits of true color information.

Hardware Graphics Cursor Storage Start Address Registers (HWGC_STA(H)(L) (CR4C, CR4D)

Read/Write Address: 3?5H, Index 4CH, 4DH
Power-On Default: Undefined

The high order four bits are written into CR4C and the low order byte is written into CR4D.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	HWGC STA(H)				HWGC STA(L)							

Bits 11–0 HWGC STA(H)(L) - Hardware Graphics Cursor Storage Start Address

Bits 15–12 Reserved

Hardware Graphics Cursor Pattern Display Start X-PXL-Position Register (HWGC_DX) (CR4E)

Read/Write Address: 3?5H, Index 4EH
Power-On Default: Undefined

7	6	5	4	3	2	1	0
R	R	HWGC PAT DISP START X-POS					

Bits 5–0 HWGC PAT DISP START X-POS - HWGC Pattern Display Start-X Pixel Position

This value is the offset (in pixels) from the left side of the 64x64 cursor pixel pattern from which the cursor is displayed. This allows a partial cursor to be displayed at the left border of the display.

Bits 7–6 Reserved

Hardware Graphics Cursor Pattern Disp Start Y-PXL-Position Register (HGC_DY) (CR4F)

Read/Write Address: 3?5H, Index 4FH
Power-On Default: Undefined

7	6	5	4	3	2	1	0
R	R	HWGC PAT DISP START Y-POS					

Bits 5–0 HWGC PAT DISP START Y-POS - HWGC Pattern Display Start-Y Pixel Position

This value is the offset (in pixels) from the top of the 64x64 cursor pixel pattern from which the cursor is displayed. This allows a partial cursor to be displayed at the top of the display.

Bits 7–6 Reserved

Extended System Control 2 Register (EX_SCTL_2) (CR51)

Read/Write Address: 3?5H, Index 51H
Power-On Default: 00H

7	6	5	4	3	2	1	0
	DIS	LOG-SCR-W		OLD-CBAD		OLD-DSAD	
R	SPXF	9	8	19	18	19	18

Bits 1–0 OLD-DSAD - Old Display Start Address Bits 19-18
These are extension bits of Memory Configuration register (CR31) bits 5-4 (Display Start Base Address). If the upper 4 display start address bits are programmed into bits 3-0 of CR69, these bits and bits 5-4 of CR31 are ignored.

Bits 3–2 OLD-CBAD - Old CPU Base Address Bits 19-18
These are extension bits of CRT Register Lock register (CR35) bits 3-0 (CPU Base Address). They becomes bits 19-18 of the CPU base address, enabling access to up to 4 MBytes of display memory. If the upper 6 CPU base address bits are programmed into bits 5-0 of CR6A, these bits and bits 3-0 of CR35 are ignored.

Bits 5–4 LOG-SCR-W - Logical Screen Width Bits 9-8
These are two extension bits of the Offset register (CR13). If the value of these bits is not 00b, bit 2 of the Extended Mode register (CR43) is disabled.

Bit 6 DIS SPXF - Disable Split Transfers
0 = Split VRAM transfers enabled
1 = Split VRAM transfers disabled

Split VRAM transfers (half the data in one SAM is transferred from the DRAM side to the serial out side while the other half is output to the RAMDAC) are required in all Enhanced modes. They are not required for VGA modes, but can be left enabled for these modes.

Bit 7 Reserved

Extended BIOS Flag 1 Register (EXT_BBFLG1) (CR52)

Read/Write Address: 3?5H, Index 52H
Power-On Default: 00H

7	6	5	4	3	2	1	0
EXT-BIOS-FLAG-REGISTER-1							

Bits 7–0 EXT-BIOS-FLAG-REGISTER-1

Reserved for use by the video BIOS.

Extended Memory Control 1 Register (EX_MCTL_1) (CR53)

Read/Write Address: 3?5H, Index 53H
Power-On Default: See Bit Descriptions

7	6	5	4	3	2	1	0
R	SWP NBL	MMIO WIN	MMIO SELECT		BIG ENDIAN LIN ADDR		ALFA PTCH

Bit 0 ALFA PTCH - Enable Alpha Pitching
0 = Alpha pitching disabled
1 = Alpha pitching enabled

Alpha pitching is available with linear addressing to a 2-MByte or larger window.

Bits 2-1 BIG ENDIAN LIN ADDR - Big Endian Data Byte swap (linear addressing only)
00 = No swap (Default)
01 = Swap bytes within each word
10 = Swap all bytes in doublewords (bytes reversed)
11 = Reserved

Bits 4-3 MMIO SELECT
00 = Disable MMIO
01 = New MMIO (relocatable) enabled (Default)
10 = Trio64-type MMIO enabled at window selected by bit 5 of this register
11 = Trio64-type MMIO and new MMIO enabled

Refer to the MMIO explanation in Section 15 for more information.

Bit 5 MMIO WIN - Trio64-type MMIO Window
0 = Trio64-type MMIO window enabled at A8000H - AFFFFH. A0000H - A7FFF available for image transfers (Default)
1 = Trio64-type MMIO window enabled at B8000H - BFFFFH. A0000H - B7FFFH are not used (no image transfer area)

Bits 4-3 of this register must be programmed to 10b for this bit to be effective.

Bit 6 SWP NBL - Swap Nibbles
0 = No nibble swap (Default)
1 = Swap nibbles in each byte of a linear memory address read or write operation

Bit 7 Reserved

Extended Memory Control 2 Register (EX_MCTL_2) (CR54)

Read/Write Address: 3?5H, Index 54H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	R	R	R	R	BIG ENDIAN	

Bits 1–0 BIG ENDIAN - Big Endian Data Byte Swap (not linear addressing or image writes)
00 = No swap (Default)
01 = Swap bytes within each word
10 = Swap all bytes in doublewords (bytes reversed)
11 = Swap according to $\overline{C/BE[3:0]}$

Byte enable settings for a bit setting of 11b:
0000 = Swap all bytes in doublewords (bytes reversed)
0011 = Swap bytes within selected word
1100 = Swap bytes within selected word
All other values = no swap

Bits 7–2 Reserved

Extended RAMDAC Control Register (EX_DAC_CT) (CR55)

Read/Write Address: 3?5H, Index 55H
Power-On Default: 00H

7	6	5	4	3	2	1	0
TOFF VCLK	CAA	R	MS /X11	ENB EDAC	R	DAC RS 3	2

Bits 1–0 DAC RS - RAMDAC Register Select Bits 3-2

These are the extension bits 3-2 of the RAMDAC register select bits. Bits 1-0 are generated via DAC register accesses as explained in Section 16.6

Bit 2 Reserved

Bit 3 ENB EDAC - Enable External RAMDAC
0 = Use internal RAMDAC
1 = Use external RAMDAC

Bit 4 MS/X11 - Hardware Cursor MS/X11 Mode
0 = MS-Windows mode (Default)
1 = X11-Windows mode

This bit select the type of decoding used for the 64x64x2 storage array of the hardware graphics cursor. See the Programming the Hardware Cursor section for a description of the decoding.

Bit 5 Reserved

Bit 6 CAA - CLUT Access Alignment
0 = 6-bit CLUT data shifted to high order bits on writes and low order bits on reads
1 = No data shift. Assumes 8-bit CLUT data.

This bit must be set to 1 when gamma correction is enabled (CR67_1 = 1).

Bit 7 TOFF VCLK - Tri-State Off VCLK Output
0 = Normal operation
1 = VCLK output is tri-stated off

External Sync Control 1 Register (EX_SYNC_1) (CR56)

Read/Write Address: 3?5H, Index 56H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	R	R	R	DIS VSYN	DIS HSYN	R

Bit 0 Reserved

Bit 1 DIS HSYN - Tri-state off HSYNC
0 = HSYNC output buffer tri-stated on
1 = HSYNC output buffer tri-stated off

Bit 2 DIS VSYN - Tri-state off VSYNC
0 = VSYNC output buffer tri-stated on
1 = VSYNC output buffer tri-stated off

Bits 7–3 Reserved

Block Write Control Register (CR57)

Read/Write
Address: 3?5H, Index 57H
Power-On Default: 00H

7	6	5	4	3	2	1	0
BWE	R	BWT	R	R	R	R	R

Bits 4-0 Reserved

Bit 5 BWT - Block Write Transfer Width
0 = 16-byte block write minimum transfer width for BitBLts
1 = 32-byte block write minimum transfer width for BitBLts

Bit 6 Reserved

Bit 7 BWE - Block Write Enable
0 = Block writes disabled
1 = Block writes enabled

Linear Address Window Control Register (LAW_CTL) (CR58)

Read/Write
Address: 3?5H, Index 58H
Power-On Default: 00H

7	6	5	4	3	2	1	0
RAS PRE	SAM 256	R	ENB LA	R	R	LAW-SIZE 1	0

Bits 1–0 LAW-SIZE - Linear Address Window Size
00 = 64 KBytes (Default)
01 = 1 MByte
10 = 2 MBytes
11 = 8 MBytes

Programmers should set these bits to 11b for 4- and 6-MByte configurations and be aware that part of the address space is not backed up with real memory.

Bits 3-2 Reserved

Bit 4 ENB LA - Enable Linear Addressing
0 = Disable linear addressing (Default)
1 = Enable linear addressing

Enabling linear addressing disables access to the A000H-AFFFH region unless banking is enabled via bit 0 of CR31, the window size is set to 64K via bits 1-0 of this register and A000H is specified as the base in CR59-5A.

This bit is ORed with MM850C_4 and is equivalent to it.

Bit 5 Reserved

Bit 6 SAM 256 - Serial Access Mode 256 Words Control
0 = SAM control is 512 words
1 = SAM control is 256 words

This setting is VRAM-dependent. A setting of 1 always works. If the VRAM can support a setting of 0, this can enhance performance.

Bit 7 RAS PRE - RAS Pre-charge Time Adjust
0 = $\overline{\text{RAS}}$ pre-charge (high) time is defined by CR68_3 or MM8204_1.
1 = $\overline{\text{RAS}}$ pre-charge (high) time is decreased by 0.5 MCLKs over that specified by CR68_3 and the corresponding $\overline{\text{RAS}}$ low time (CR68_2) is increased by 0.5 MCLKs. This leaves the total cycle time unchanged.

Linear Address Window Position Registers (LAW_POS(X) (CR59-5A)

Read/Write Address: 3?5H, Index 59H-5AH
Power-On Default: 7000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LINEAR-ADDRESS-WINDOW-POSITION															

CR59 contains the upper byte (15-8) and CR5A contains the lower byte (7-0). These registers specify system address bits 31-16 of the Linear Address Window Position in 32-bit CPU address space. The Linear Address Window resides on a 64KB, 1MB, 2MB or 8MB memory boundary (size-aligned boundary). Some LSBs of this register (illustrated by "xx..xx" in the following table) are ignored because of the size-aligned boundary scheme.

LAW Size	Linear Address Window Position Register Bit(s)									
64KB	31-25	24	23	22	21	20	19	18	17	16
1MB	31-25	24	23	22	21	20	xx	xx	xx	xx
2MB	31-25	24	23	22	21	xx	xx	xx	xx	xx
8MB	31-25	24	23	xx	xx	xx	xx	xx	xx	xx

Bits 15–0 LINEAR-ADDRESS-WINDOW-POSITION - LA Window Position Bits 31-16
16-bit Value = the linear address window position in 32-bit CPU address space.

Bits 15-0 are common with bits 31-16 of the base address programmed into the PCI Base Address 0 register at address 10H-12H. Writes to these bits in either register will also be written to the other. Writes to CR59 and CR5A should be read-modify- writes that do not change the upper 6 bits, as these bits are written by the system BIOS to place ViRGE/VX in a unique address space. Note that system BIOS writes will leave bits 9-0 in an indeterminate state, so these should be properly initialized before linear addressing is enabled.

If a 64K window is specified and bit 0 of CR31 is set to 1, bits 6-0 of CR6A specify the 64K page of display memory to be accessed through a 64K window located at the address specified in these registers.

General Output Port Register (GOUT_PORT) (CR5C)

Read/Write Address: 3?5H, Index 5CH
Power-On Default: 00H

7	6	5	4	3	2	1	0
GENERAL OUT PORT							

Bits 7–0 GENERAL OUT PORT

This register can be used in a variety of ways. See Section 12-4 for a complete description.

Extended Horizontal Overflow Register (EXT_H_OVF) (CR5D)

Read/Write Address: 3?5H, Index 5DH
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	DTP 8	EHS 6	SHS 8	EHB 7	SHB 8	HDE 8	HT 8

Bit 0 HT 8 - Horizontal Total (CR0) Bit 8

Bit 1 HDE 8 - Horizontal Display End (CR1) Bit 8

Bit 2 SHB 8 - Start Horizontal Blank (CR2) Bit 8

Bit 3 EHB 7 - End Horizontal Blank (CR3) Bit 6

Bit 4 SHS 8 - Start Horizontal Sync Position (CR4) Bit 8

Bit 5 EHS 6 - End Horizontal Sync (CR5) Bit 6

Bit 6 SFF 8 - Data Transfer Position (CR3B) Bit 8

Bit 7 Reserved

Extended Vertical Overflow Register (EXT_V_OVF) (CR5E)

Read/Write Address: 3?5H, Index 5EH
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	LCM 10	R	VRS 10	R	SVB 10	VDE 10	VT 10

Bit 0 VT 10 - Vertical Total (CR6) Bit 10

Bit 1 VDE 10 - Vertical Display End (CR12) Bit 10

Bit 2 SVB 10 - Start Vertical Blank (CR15) Bit 10

Bit 3 Reserved

Bit 4 VRS 10 - Vertical Retrace Start (CR10) Bit 10

Bit 5 Reserved

Bit 6 LCM 10 - Line Compare Position (CR18) Bit 10

Bit 7 Reserved

Extended Memory Control 4 Register (EXT-MCTL-4) (CR61)

Read/Write Address: 3?5H, Index 61H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	BIG ENDIAN		R	R	R	R	R

Bits 4-0 Reserved

Bits 6-5 BIT ENDIAN - Big Endian Data Byte Swap (image writes only)
00 = No swap (Default)
01 = Swap bytes within each word
10 = Swap all bytes in doublewords (bytes reversed)
11 = Reserved

Bit 7 Reserved

Extended Control Register (CR63)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 63H

7	6	5	4	3	2	1	0
DELAY HSYNC/VSYNC				PCI DISC	R	RST	ENBL ENH

Bit 0 ENBL ENH - Enable Enhanced Functions
0 = Enable VGA and VESA planar (4 bits/pixel) modes
1 = Enable all other modes (Enhanced and VESA non-planar)

This bit has the same function as MM850C_0. It enables operation of the S3d Engine.

Bit 1 RST - Reset
0 = No operation
1 = Software reset of S3D Engine and memory controller

Setting this bit has the same effect as setting MM8504_15-14 (Write) to 10b.

Bit 2 Reserved

Bit 3 PCI DISC - PCI Disconnects
0 = No effect
1 = An attempt to write data with the Command FIFO or LPB Output FIFO full or to read data with the Command FIFO not empty generates a PCI bus disconnect cycle

Bit 7 of CR66_7 must also be set to 1 to enable this feature.

Bits 7-4 DELAY HSYNC/VSYNC

value = number of DCLKs the HSYNC and VSYNC active pulses are delayed

Extended Miscellaneous Control Register (EXT-MISC-CTL) (CR65)

Read/Write Address: 3?5H, Index 65H
Power-On Default: 00H

7	6	5	4	3	2	1	0
ADR ADJ		R	R	R	R	R	SED

Bit 0 SED - Delay Falling Edge of SE
0 = Normal operation
1 = Delay falling edges of $\overline{SE[3:0]}$

The amount of delay is approximately a few nanoseconds.

Bits 5-1 Reserved

Bits 7-6 ADR ADJ - Address Adjustments for Split Transfers
00 = 64-bit SID bus with 256 word SAMs
01 = Reserved
10 = 64-bit SID bus with 512 word SAMs or 128-bit SID bus with 256 word SAMs
11 = 128-bit SID bus with 512 word SAMs

These bits must be set in a manner consistent with CR66_5-4 and CR58_6.

Extended Miscellaneous Control 1 Register (EXT-MISC-1) (CR66)

Read/Write Address: 3?5H, Index 66H
Power-On Default: 00H

7	6	5	4	3	2	1	0
PCI DE	TOFF PADT	SID MODE		INV SC	DIV SC, SE, ICLK		

Bits 2-0 DIV SC, SE, ICLK
000 SC[3:0], $\overline{SE[3:0]}$ and ICLK = DCLK
001 SC[3:0], $\overline{SE[3:0]}$ and ICLK = DCLK/2
010 SC[3:0], $\overline{SE[3:0]}$ and ICLK = DCLK/4
011 SC[3:0], $\overline{SE[3:0]}$ and ICLK = DCLK/8
100 SC[3:0], $\overline{SE[3:0]}$ and ICLK = DCLK/16
101 SC[3:0], $\overline{SE[3:0]}$ and ICLK = DCLK/32
110 = Reserved
111 = Reserved

Bit 3 INV SC - Invert SC
0 = Normal operation
1 = SC[3:0] signals are inverted

Bits 5-4 SID MODE - SID Operation Mode
00 = 64-bit serial SID bus
01 = 64-bit serial SID Type B (4-MByte option for 128-bit configuration)
10 = 128-bit parallel SID bus
11 = Reserved

Bit 6 TOFF PADT - Tri-State Off Pixel Address Bus
0 = Normal operation
1 = PA[15:0] are set to tri-state off

Bit 7 PCI DE - PCI bus disconnect enable
0 = PCI bus disconnect disabled
1 = PCI bus disconnect enabled

Setting this bit to 1 allows PCI burst cycles to be interrupted if AD[1:0] ≠ 00b or if the address during the burst goes outside the address ranges supported by ViRGE/VX.

Extended Miscellaneous Control 2 Register (EXT-MISC-2)(CR67)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 67H

7	6	5	4	3	2	1	0
COLOR MODE				STREAMS MODE		ENB GC	VCLK PHS
3	2	1	0				

Bit 0 ICLK PHS - ICLK Phase With Respect to DCLK
0 = ICLK is 180° out of phase with DCLK (inverted)
1 = ICLK is in phase with DCLK

Bit 1 ENB GC - Enable Gamma Correction
0 = Gamma correction disabled
1 = Gamma correction enabled

CR55_6 must be set to 1 when gamma correction is enabled.

Bits 3–2 STREAMS MODE
00 = Streams Processor disabled
01 = Secondary stream overlaid on VGA mode background
10 = Reserved
11 = Full Streams Processor operation (primary and secondary streams from all supported sources)

The Streams Processor should only be enabled or disabled during the VSYNC period.

Bits 7–4 COLOR MODE - RAMDAC Color Mode
0000 = 8-bit color; 135 MHz maximum DCLK
0001 = 8-bit color; 220 MHz maximum DCLK
0010 = 15-bit color; 135 MHz maximum DCLK
0011 = 15-bit color; 220 MHz maximum DCLK
0100 = 16-bit color; 135 MHz maximum DCLK
0101 = 16-bit color; 220 MHz maximum DCLK
1101 = 24-bit color; 135 MHz maximum DCLK

All other mode values are reserved. 220 MHz is recommended only for 1600x1200 modes, for which the Streams Processor should not be turned on. The maximum operating frequency of the Streams Processor is 135 MHz.

Configuration 3 Register (CNFG-REG-3) (CR68)

Read/Write Address: 3?5H, Index 68H
Power-On Default: Depends on Strapping

This is the third byte (along with CR36 and CR37) of the power-on strapping bits. CR6F contains the fourth byte. PD[23:16] are sampled on power-on reset and their states are written to bits 7-0 of this register. A5H must be written to CR39 to provide read/write access to this register.

7	6	5	4	3	2	1	0
R	BIOS AREA			RAS - PCG	RAS - LOW	CAS LE	CAS TE

Bits 1-0 $\overline{CAS}$,$\overline{OE}$ ADJUST
00 = approximately 6.5 ns adjustment (nominal)
01 = approximately 5 ns adjustment (nominal)
10 = approximately 3.5 ns adjustment (nominal)
11 = no adjustment

If MM8204_7 = 0, the trailing edges of $\overline{CAS}$/$\overline{OE}$ active are delayed by the amount indicated by these bits. If MM8204_7 = 1, the entire $\overline{CAS}$/$\overline{OE}$ active pulses are shifted (delayed) by the amount indicated by these bits. The timing adjustment shown above is an approximation. It is affected by both process and signal loading and must be measured for each design.

Bit 2 $\overline{RAS}$-LOW - $\overline{RAS}$ Low Timing Select
0 = 4.5 MCLKs
1 = 3.5 MCLKs

This parameter specifies the length of the $\overline{RAS}$ active time for a single row/column access. $\overline{RAS}$ may be held low longer to accommodate additional page mode accesses to the same row.

Bit 3 $\overline{RAS}$-PCG - $\overline{RAS}$ Precharge Timing Select
0 = 3.5 MCLKs
1 = 2.5 MCLKs

When $\overline{RAS}$ goes high to end a memory cycle, this parameter specifies the minimum period it must be held high before beginning another memory access cycle.

Bits 6-4 BIOS AREA

Reserved for use by the video BIOS.

Bit 7 Reserved

Extended System Control 3 Register (EXT-SCTL-3)(CR69)

Read/Write Address: 3?5H, Index 69H
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	R	R	DISPLAY-START-ADDRESS				

Bits 4–0 DISPLAY-START-ADDRESS
This field contains the upper 5 bits (20-16) of the display start address, allowing addressing of up to 8 MBytes of display memory. When a non-zero value is programmed in this field, bits 5-4 of CR31 and 1-0 of CR51 (the old display start address bits) are ignored.

Bits 7–5 Reserved

Extended System Control 4 Register (EXT-SCTL-4)(CR6A)

Read/Write Address: 3?5H, Index 6AH
Power-On Default: 00H

7	6	5	4	3	2	1	0
R	CPU-BASE-ADDRESS						

Bits 6–0 CPU-BASE-ADDRESS
This field contains the upper 7 bits (20-14) of the CPU base address, allowing accessing of up to 8 MBytes of display memory via 64K pages. When a non-zero value is programmed in this field, bits 3-0 of CR35 and 3-2 of CR51 (the old CPU base address bits) are ignored. Bit 0 of CR31 must be set to 1 to enable this field. If linear addressing is enabled and a 64 KByte window is specified, these bits specify the 64K page to be accessed at the base address specified in CR59 and CR5A. Otherwise, the base address is normally at A000H.

Bit 7 Reserved

Extended BIOS Flag 3 Register (EBIOS-FLG3)(CR6B)

Read/Write Address: 3?5H, Index 6BH
Power-On Default: 00H

7	6	5	4	3	2	1	0
EXT-BIOS-FLAG-REGISTER-3							

Bits 7–0 EXT-BIOS-FLAG-REGISTER-3
This register is reserved for use by the S3 BIOS.

Extended BIOS Flag 4 Register (EBIOS-FLG4)(CR6C)

Read/Write Address: 3?5H, Index 6CH
Power-On Default: 00H

7	6	5	4	3	2	1	0
EXT-BIOS-FLAG-REGISTER-4							

Bits 7–0 EXT-BIOS-FLAG-REGISTER-4
This register is reserved for use by the S3 BIOS.

Signal Delay Register (CR6D)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 6DH

7	6	5	4	3	2	1	0
DELAY SC				DELAY BLANK			

Bits 3–0 DELAY BLANK

value = number of ICLKs by which assertion of the $\overline{\text{BLANK}}$ signal is delayed

Bits 7-4 DELAY SC

value = number of ICLKs by which SC{1:0] are delayed

RAMDAC Signature Data Register (CR6E)

Read/Write
Power-On Default: 00H

Address: 3?5H, Index 6EH

7	6	5	4	3	2	1	0
RAMDAC SIGNATURE DATA							

Bits 7–0 RAMDAC SIGNATURE DATA

See Section 8.10 for a description of how to use this register.

Configuration 4 Register (CR6F)

See Bit Definitions Address: 3?5H, Index 6FH
Power-On Default: Depends on Strapping

This is the fourth byte of power-on strapping bits. PD[28:24] are sampled at reset and the values are written to bits 4-0 of this register. A5H must be written to CR39 to provide read/write access to this register. This register will power up with a value of 1FH if any of PD[28:24] are not pulled low.

7	6	5	4	3	2	1	0
R	R	R	WE DELAY		IOEN	IOSEL	R

Bit 0 Reserved

Bit 1 IOSEL - Serial Port I/O Address Select (read/write)
0 = Serial Port register is accessed at I/O address 000E8H
1 = Serial Port register is accessed at I/O address 000E2H

Bit 2 of this register must be cleared to 0 for this bit to have effect.

Bit 2 IOEN - Serial Port Address Type Select (read/write)
0 = Serial Port register is accessed at the I/O port defined in bit 1 of this register or at its MMIO address (offset FF20H)
1 = Serial Port register is accessed at its MMIO address only (offset FF20H)

Enabling I/O access allows the serial port to be used for I^2C communications when ViRGE/VX is disabled.

Bits 4-3 $\overline{WE}$ Delay (read/write)
00 = 3 units delay
01 = 2 units delay
10 = 1 unit delay
11 = 0 units delay

If MM8204_8 = 0, both the leading and trailing edges of $\overline{WE}$ are delayed by the amount specified in these bits. If MM8204_8 = 1, only the trailing edge is delayed by the amount specified in these bits.

Bits 7-5 Reserved

Signal Drive Strength Register (CR80)

Read/Write Address: 3?5H, Index 80H
Power-On Default: 00H

7	6	5	4	3	2	1	0
SE	SC	PD	MA	OE	DSF	CAS	RAS

Bit 0 RAS Drive Strength
0 = $\overline{\text{RAS[1:0]}}$ I_{OL} = 8 mA
1 = $\overline{\text{RAS[1:0}}$} I_{OL} = 16 mA

Bit 1 CAS Drive Strength
0 = $\overline{\text{CAS[7:0]}}$ I_{OL} = 4 mA
1 = $\overline{\text{CAS[7:0}}$} I_{OL} = 8 mA

Bit 2 DSF Drive Strength
0 = DSF[2:0] I_{OL} = 8 mA
1 = DSF[2:0} I_{OL} = 16 mA

Bit 3 OE/WE Drive Strength
0 = $\overline{\text{OE[1:0]/WE[1:0]}}$ I_{OL} = 8 mA
1 = $\overline{\text{OE[1:0]/WE[1:0]}}$ I_{OL} = 16 mA

Bit 4 MA Drive Strength
0 = MA[8:0] I_{OL} = 8 mA
1 = MA[8:0} I_{OL} = 16 mA

Bit 5 PD Drive Strength
0 = PD[63:0] I_{OL} = 4 mA
1 = PD[63:0} I_{OL} = 8 mA

Bit 6 SC Drive Strength
0 = SC[1:0] I_{OL} = 8 mA
1 = SC[1:0} I_{OL} = 16 mA

Bit 7 SE Drive Strength
0 = $\overline{\text{SE[1:0]}}$ I_{OL} = 8 mA
1 = $\overline{\text{SE[1:0}}$} I_{OL} = 16 mA

Section 19: S3d Engine Register Descriptions

This section describes the S3d Registers for ViRGE/VX. These registers are used to accelerate the display of 2D and 3D graphics.

In all register bit descriptions, the letter "R" identifies reserved bits (a reserved bit's read value is undefined unless noted, and you may write only zero to a reserved bit).

19.1 REGISTER MAPPING AND ADDRESSING

The S3d registers are memory-mapped starting at an offset of 100 A0000H from the base address. Table 20-1 shows the location of each register organized by drawing command type. All registers with the same mnemonic for different commands are the same register with multiple addresses. For example, at "xx" = D4, the three 2D commands use a register called SRC_BASE, with each of the 2D commands having a unique address for this register. Similarly, the two 3D commands share the Z-BASE register. The DEST_BASE register is shared by all commands at "xx" = D8. Each shared register is described only once in a section (2D or 3D) along with all of its addresses.

Table 19-1. S3d Register Memory Map

	Offset From Base Address (Little Endian Addressing)					
	100 A0xxH	100 A4xxH	100 A8xxH	100 ACxxH	100 B0xxH	100 B4xxH
xx	Pattern Registers	BitBLT/Rect Fill	2D Line	2D Polygon	3D Line	3D Triangle
D4		SRC_BASE	SRC_BASE	SRC_BASE	Z_BASE	Z_BASE
D8		DEST_BASE	DEST_BASE	DEST_BASE	DEST_BASE	DEST_BASE
DC		CLIP_L_R	CLIP_L_R	CLIP_L_R	CLIP_L_R	CLIP_L_R
E0		CLIP_T_B	CLIP_T_B	CLIP_T_B	CLIP_T_B	CLIP_T_B
E4		DEST_SRC_STR	DEST_SRC_STR	DEST_SRC_STR	DEST_SRC_STR	DEST_SRC_STR
E8		MONO_PAT_0		MONO_PAT_0	Z_STRIDE	Z_STRIDE
EC		MONO_PAT_1		MONO_PAT_1		TEX_BASE
F0		PAT_BG_CLR		PAT_BG_CLR		TEX_BDR_CLR
F4		PAT_FG_CLR	PAT_FG_CLR	PAT_FG_CLR	FOG_CLR	FOG_CLR
F8		SRC_BG_CLR				COLOR0
FC		SRC_FG_CLR				COLOR1
100	Start	CMD_SET	CMD_SET	CMD_SET	CMD_SET	CMD_SET
104	(100 to 1BC)	RWIDTH_HEIGHT				TBV
108		RSRC_XY				TBU
10C		RDEST_XY				TdWdX
110						TdWdY
114						TWS
118						TdDdX
11C						TdVdX
120						TdUdX
124						TdDdY
128						TdVdY
12C						TdUdY
130						TDS
134						TVS
138						TUS
13C						TdGdX_dBdX
140						TdAdX_dRdX
144					3dGdY_dBdY	TdGdY_dBdY
148					3dAdY_dRdY	TdAdY_dRdY
14C					3GS_BS	TGS_BS
150					3AS_RS	TAS_RS
154						TdZdX
158					3dZ	TdZdY
15C					3ZSTART	TZS
160						TdXdY12
164						TXEND12
168				PRdX		TdXdY01
16C			LXEND0_END1	PRXSTART	3XEND0_END1	TXEND01
170			LdX	PLdX	3dX	TdXdY02
174			LXSTART	PLXSTART	3XSTART	TXSTART02
178			LYSTART	PYSTART	3YSTART	TYSTART
17C			LYCNT	PYCNT	3YCNT	TY_01_Y12

19.2 COLOR PATTERN REGISTERS

When the ROP chosen for a BitBLT uses a color pattern, the 8x8 pixel pattern data must be stored in the register address space starting at offset 100 A100H. The amount of register space required is a function of the color depth as shown in Table 20-2. The value is derived by multiplying 64 pixels (8x8 pattern) by the color depth (bytes/pixel) and dividing by 4 bytes/doubleword (32-bit registers).

Table 19-2 Color Pattern Data Storage Requirements

Color Depth (Bits/Pixel)	Storage Requirements (Doublewords)	Offset Range (Hex)
8	16	100 A100 - 100 A13C
16	32	100 A100 - 100 A17C
24	48	100 A100 - 100 A1BC

The pattern color data is written starting with the upper left pixel (0,0) to the end of the line (7,0) and then proceeding across each line to the last pixel (8,8). Pixel 0,0 is written to 100 A100H. The data are stored fully packed.

For 8 bits/pixel, pixel 0,0 is written to the low order byte 0, pixel 1,0 is written to byte 1, etc. Pixel 4,0 would then be written to the low order byte of 100 A104H and so on. The 8-bit value for each pixel is an index to the DAC palette registers.

For 16 bits/pixel, pixel 0,0 is written to the low order word of 100 A100H, pixel 1,0 to the high order word, etc. Either RGB1555 or RGB565 coding can be used.

For 24 bits/pixel, pixel 0,0 is written to the 3 low order bytes of 100 A100H (RGB888 format). The blue value for pixel 1,0 is written to the high order byte of 100 A100H. The red and green values for pixel 1,0 are written to the low order word of 100 A104H and so on. Thus pixel data crosses doubleword boundaries.

19.3 2D REGISTERS

This section describes all the registers used with the 2D drawing commands (BitBLT/Rectangle Fill, 2D Line and 2D Polygon).

Source Base Address Register (SRC_BASE) (MMA4D4, MMA8D4, MMACD4)

Read/Write Offset: A4D4H (BitBLT), A8D4H (2D Line), ACD4H (2D Polygon)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOURCE BASE ADDRESS													0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	SOURCE BASE ADDRESS					

Bits 2-0 Reserved = 0

Bits 21-3 SOURCE BASE ADDRESS

Value = base address in video memory of source data for 2D drawing operations (quadword aligned)

This value is required when the source is video memory (screen). It is different from the destination base address when the data is located in off-screen memory. This is the 0,0 pixel address for off-screen data. The stride for off-screen data is programmed in the Destination/Source Stride register (MMxxE4).

Bits 31-22 Reserved

Destination Base Address Register (DEST_BASE) (MMA4D8, MMA8D8, MMACD8)

Read/Write Offset: A4D8H (BitBLT), A8D8H (2D Line), ACD8H (2D Polygon)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DESTINATION BASE ADDRESS													0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	DESTINATION BASE ADDRESS					

Bits 2-0 Reserved = 0

Bits 21-3 DESTINATION BASE ADDRESS

Value = base address in video memory of destination data for 2D drawing operations (quadword aligned)

This is the 0,0 pixel address in video memory for the screen resolution being used. It will normally be at the start of video memory.

Bits 31-22 Reserved

Left/Right Clipping Register (CLIP_L_R) (MMA4DC, MMA8DC, MMACDC)

Read/Write Offset: A4DCH (BitBLT), A8DCH (2D Line), ACDCH (2D Polygon)
Power-On Default: Undefined

Bit 1 of the Command Set register must be set to 1 for the settings in this register to have effect.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	RIGHT CLIPPING LIMIT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	LEFT CLIPPING LIMIT										

Bits 10-0 RIGHT CLIPPING LIMIT

Value = pixel position of the last pixel to be drawn on each line. The first pixel is 0.

Bits 15-11 Reserved

Bits 26-16 LEFT CLIPPING LIMIT

Value = pixel position of the first pixel to be drawn on each line. The first pixel is 0.

Bits 31-27 Reserved

Top/Bottom Clipping Register (CLIP_T_B) (MMA4E0, MMA8E0, MMACE0)

Read/Write Offset: A4E0H (BitBLT), A8E0H (2D Line), ACE0H (2D Polygon)
Power-On Default: Undefined

Bit 1 of the Command Set register must be set to 1 for the settings in this register to have effect.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	BOTTOM CLIPPING LIMIT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	TOP CLIPPING LIMIT										

Bits 10-0 BOTTOM CLIPPING LIMIT

Value = line position of the last line to be drawn. The first line is 0.

Bits 15-11 Reserved

Bits 26-16 TOP CLIPPING LIMIT

Value = line position of the first line to be drawn. The first line is 0.

Bits 31-27 Reserved

Destination/Source Stride Register (DEST_SRC_STR) (MMA4E4, MMA8E4, MMACE4)

Read/Write Offset: A4E4H (BitBLT), A8E4H (2D Line), ACE4H (2D Polygon)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	SOURCE STRIDE									0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	DESTINATION STRIDE									0	0	0

Bits 11-0 SOURCE STRIDE

Value = byte offset of vertically adjacent pixels for the source data. Bits 2-0 must be 000b.

Bits 15-12 Reserved

Bits 27-16 DESTINATION STRIDE

Value = byte offset of vertically adjacent pixels for the destination data. Bits 2-0 must be 000b.

Bits 31-28 Reserved

Mono Pattern 0 Register (MONO_PAT_0) (MMA4E8, MMACE8)

Read/Write Offset: A4E8H (BitBLT), ACE8H (2D Polygon)
Power-On Default: Undefined

The pattern data in this register is used when bit 8 of the Command Set register is set to 1 to specify a mono pattern. The first four lines of the pattern are specified in this register.

15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
L20	L21	L22	L23	L24	L25	L26	L27	L10	L11	L12	L13	L14	L15	L16	L17
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
L40	L41	L42	L43	L44	L45	L46	L47	L30	L31	L32	L33	L34	L35	L36	L37

Bits 31-0 MONO PATTERN 0

Value = first (low order) 32 bits of a 64-bit mono pattern

The second (high order) 32 bits are found in the Mono Pattern 1 register. These two registers define an 8x8 mono pattern. In the above register bit table, LXY means bit Y of line X, with the leftmost bit of each line (row) being bit 0.

Mono Pattern 1 Register (MONO_PAT_1) (MMA4EC, MMACEC)

Read/Write Offset: A4ECH (BitBLT), ACECH (2D Polygon)
Power-On Default: Undefined

The pattern data in this register is used when bit 8 of the Command Set register is set to 1 to specify a mono pattern. The second four lines of the pattern are specified in this register.

15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
L60	L61	L62	L63	L64	L65	L66	L67	L50	L51	L52	L53	L54	L55	L56	L57
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
L80	L81	L82	L83	L84	L85	L86	L87	L70	L71	L72	L73	L74	L75	76	L77

Bits 31-0 MONO PATTERN 1

Value = second (high order) 32 bits of a 64-bit mono pattern (little endian format)

The first (low order) 32 bits are found in the Mono Pattern 0 register. These two registers define an 8x8 mono pattern. In the above register bit table, LXY means bit Y of line X, with the leftmost bit of each line (row) being bit 0.

Mono Pattern Background Color Register (PAT_BG_CLR) (MMA4F0, MMACF0)

Read/Write Offset: A4F0H (BitBLT), ACF0H (2D Polygon)
Power-On Default: Undefined

The pattern color data in this register is used when bit 8 of the Command Set register is set to 1 to specify a mono pattern and the pattern bit is 0. The color depth specified must match the value selected by bits 4-2 of the Command Set register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = DAC CLUT index (8 bits/pixel), lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel

Bits 15-8 DATA 2

Value = Reserved (8 bits/pixel), upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

Bit 23-16 DATA 3

Value = Reserved (8, 15 or 16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Mono Pattern Foreground Color Register (PAT_FG_CLR) (MMA4F4, MMA8F4, MMACF4)

Read/Write Offset: A4F4H (BitBLT), A8F4H (2D Line), ACF4H (2D Polygon)
Power-On Default: Undefined

The pattern color data in this register is used when bit 8 of the Command Set register is set to 1 to specify a mono pattern and the pattern bit is 1. It is also the pattern color used for rectangle fills, line draws and polygon fills, regardless of any pattern specification. The color depth specified must match the value selected by bits 4-2 of the Command Set register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = DAC CLUT index (8 bits/pixel), lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel

Bits 15-8 DATA 2

Value = Reserved (8 bits/pixel), upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

Bit 23-16 DATA 3

Value = Reserved (8, 15 or 16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Source Background Color Register (SRC_BG_CLR) (MMA4F8)

Read/Write Offset: A4F8H (BitBLT)
Power-On Default: Undefined

For mono image transfers (bit 6 of the Command Set register set to 1), this is the source color when the image bit is 0. It is not used when color compare is enabled (bit 9 of the Command Set register set to 1). The color depth specified must match the value selected by bits 4-2 of the Command Set register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = DAC CLUT index (8 bits/pixel), lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel

Bits 15-8 DATA 2

Value = Reserved (8 bits/pixel), upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

Bit 23-16 DATA 3

Value = Reserved (8, 15 or 16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Source Foreground Color Register (SRC_FG_CLR) (MMA4FC)

Read/Write Offset: A4FCH (BitBLT)
Power-On Default: Undefined

For mono image transfers (bit 6 of the Command Set register set to 1), this is the source color when the image bit is 1. For 8- or 15/16-bits/pixel color image transfers when transparent color is enabled (bit 9 of the Command Set register set to 1), the image data color is compared with this color. If it matches, the screen is not updated. If it does not match, the image data color is used to update the screen. In all cases, the color depth specified must match the value selected by bits 4-2 of the Command Set register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = DAC CLUT index (8 bits/pixel), lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel)

The 24 bits/pixel color is used only for mono image transfers.

Bits 15-8 DATA 2

Value = DAC CLUT index (8 bits/pixel), upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

The 8 bits/pixel color must be programmed to both the DATA 1 and DATA 2 bytes. The 24 bits/pixel color is used only for mono image transfers.

Bit 23-16 DATA 3

Value = Reserved (8, 15 or 16 bits/pixel), red color index (24 bits/pixel)

This color is used only for mono image transfers.

Bits 31-24 Reserved

Command Set Register (CMD_SET) (MMA500, MMA900, MMAD00)

Read/Write Offset: A500H (BitBLT), A900H (2D Line), AD00H (2D Polygon)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	FDO		ITA		TP	MP	IDS	MS	DE	DEST FORMAT			HC	AE
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
23D	2D COMMAND				YP	XP	256 ROPS								BWD

Bit 0 AE - Autoexecute
0 = Execute command when this register is written to
1 = Execute command when the highest address register in a drawing type set is written to

The highest address register in a drawing type set is easily seen in Table 20-1, where it is the bottom register in each column. For example, if this bit is set to 1, a BitBLT is executed when the RDEST_XY (MMA50C) register is written to. Similarly, execution of a 2D line command is based on writing to the LYCNT register, etc. This setting allows multiple executions of a given command using different parameters without rewriting the Command Set register.

To turn off autoexecute without executing a command, write to this register with this bit cleared to 0 and bits 30-27 programmed to 1111b (NOP).

Bit 1 HC - Hardware Clipping Enable
0 = Hardware clipping disabled
1 = Hardware clipping enabled

The settings in the clipping registers (MMxxDC, MMxxE0) are effective only when this bit is set to 1.

Bits 4-2 DEST FORMAT - Destination Color Format
000 = 8 bits/pixel palettized
001 = 16 bits/pixel (RGB1555 or RGB565)
010 = 24 bits/pixel, RGB888

All other values are reserved.

Bit 5 DE - Draw Enable
0 = Don't update screen
1 = Update screen (normal draw)

Parameter values calculated during the execution of the command end up the same regardless of the setting of this bit. That is, the command is fully executed except for the possible non-drawing of the new pixel.

Bit 6 MS - Mono Source (Image Transfers)
0 = Source data is the same pixel depth as the destination data
1 = Source data is mono

Bit 7 IDS - Image Data Source
0 = Source data is from video memory (screen)
1 = Source data is from the image transfer port (CPU, system memory)

When this bit is set to 1, source data is provided by CPU writes to the offset range of 100 0000H to 100 7FFFH or the alternate image transfer port range of 100 D000H to 100 EFFFH. Bit 6 of this register specifies whether mono or color data is being transferred.

Bit 8 MP - Mono Pattern
0 = Pattern data is the same pixel depth as the destination data
1 = Pattern data is mono

This bit is cleared to 0 for a BitBLT using a ROP with a color source. The 8x8 color pattern is found starting at location 100 A100H. For a mono pattern, the pattern information is determined from the Mono Pattern 0 and 1 registers. This bit must be set to 1 for a rectangle fill operation.

Bit 9 TP - Transparent
0 = A mono source image transfer uses both the source foreground (image bit = 1) and source background (image bit = 0) colors to update the screen. A color image transfer uses the CPU-provided colors.
1 = A mono source image transfer updates the screen only when the source foreground color is selected (image bit = 1). Otherwise (image bit = 0), the screen pixel is left unchanged. A color image transfer updates the screen with the transmitted color only when that color does not match the color in the source foreground color register. If a color match occurs, the destination pixel is not updated. This transparent color feature for color image transfers can be used for 8- and 16-bit color modes, but not for 24-bit color.

Note: This bit is effective only when bit 7 of this register is set to 1. A setting of 1 for the mono source case provides "transparent text" capability. The term "transparent text" refers to the updating of only the pixels forming the text characters and not the entire rectangular text block using the background color for non-text areas.

Bits 11-10 ITA - Image Transfer Alignment
00 = Data for each line of an image transfer is byte aligned
01 = Data for each line of an image transfer is word aligned
10 = Data for each line of an image transfer is doubleword aligned
11 = Reserved

All image transfers are doublewords. If the end of a bit map line is reached within a doubleword transfer, the setting of these bits determines how the start of the next line is handled. If doubleword aligned, data in the last doubleword beyond the end of the line is discarded and the next line begins on the next doubleword. If word aligned and an upper word of data remains after the end of the line is reached, that word will be used to begin the next line. If byte aligned, the next line with begin on the next byte in the doubleword after the end of the line. The latter is used only for mono source data, e.g., text.

Bits 13-12 FDO - First Doubleword Offset (Image Transfers)
00 = Entire first doubleword of an image transfer contains valid data
01 = Start with the second byte of the first doubleword of an image transfer
10 = Start with the third byte of the first doubleword of an image transfer
11 = Start with the fourth byte of the first doubleword of an image transfer

Bits 15-14 Reserved

Bit 16 BWD - Block Write Disable
0 = Enable block writes for this command if CR57_7 = 1
1 = Disable block writes for this command

This bit should be set to 1 for color pattern 8 bits/pixel BitBLTs with a transfer width of less than 64 bytes.

Bits 24-17 256 ROPS - 256 Raster Operations

Value = binary key selecting one of 256 three operand raster operations as defined in Appendix A.

The full 256 three-operand ROPs are available for BitBLT and image transfer operations. The other 2D operations (Rectangle Fill, Line Draw and Polygon Fill) can only use the subset of the 256 ROPs that does not have a source. When the ROP contains a pattern, the pattern must be mono and the hardware forces the pattern value to the pattern foreground color regardless of the values programmed in the Mono Pattern registers.

Bit 25 XP - X Positive (BitBLT)
0 = A BitBLT is performed from right to left (X negative)
1 = A BitBLT is performed from left to right (X positive)

Bit 26 YP - Y Positive (BitBLT)
0 = A BitBLT is performed from bottom to top (Y negative)
1 = A BitBLT is performed from top to bottom (Y positive)

Bits 30-27 2D COMMAND
0000 = BitBLT
0001 = Reserved
0010 = Rectangle Fill
0011 = Line Draw
0100 = Reserved
0101 = Polygon Fill
0110 = Reserved
0111 = Reserved
1000 = Reserved
1001 = Reserved
1010 = Reserved
1011 = Reserved
1100 = Reserved
1101 = Reserved
1110 = Reserved
1111 = NOP

The NOP option is required to turn off autoexecute without executing a command. See the definition for bit 0 of this register.

Bit 31 23D - 2D or 3D Select
0 = A 2D command is being executed
1 = A 3D command is being executed

Rectangle Width/Height Register (RWIDTH_HEIGHT) (MMA504)

Read/Write Offset: A504H (BitBLT)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	RECTANGLE HEIGHT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	RECTANGLE WIDTH										

Bits 10-0 RECTANGLE HEIGHT

Value = height in lines of the rectangle to be drawn or blitted

A value of 1 equals 1 line.

Bits 15-11 Reserved

Bits 26-16 RECTANGLE WIDTH

Value = width in pixels of the rectangle to be drawn or blitted

A value of 0 equals 1 pixel/line.

Bits 31-27 Reserved

Rectangle Source XY Register (RSRC_XY) (MMA508)

Read/Write Offset: A508H (BitBLT)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SOURCE Y										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	SOURCE X										

Bits 10-0 SOURCE Y

Value = y coordinate in lines of the upper left hand corner of the source rectangle for a BitBLT

Bits 15-11 Reserved

Bits 26-16 SOURCE X

Value = x coordinate in pixels of the upper left hand corner of the source rectangle for a BitBLT

Bits 31-27 Reserved

Note: The starting coordinate is 0,0.

Rectangle Destination XY Register (RDEST_XY) (MMA50C)

Read/Write Offset: A50CH (BitBLT)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	DESTINATION Y										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	DESTINATION X										

Bits 10-0 DESTINATION Y

Value = y coordinate in lines of the upper left hand corner of the filled rectangle to be drawn or the destination for a BitBLT

Bits 15-11 Reserved

Bits 26-16 DESTINATION X

Value = x coordinate in pixels of the upper left hand corner of the filled rectangle to be drawn or the destination for a BitBLT

Bits 31-27 Reserved

Note: The starting coordinate is 0,0.

Line Draw Endpoints Register (LXEND0_END1) (MMA96C)

Read/Write Offset: A96CH (2D Line)
Power-On Default: Undefined

This register specifies the x coordinates of the first and last pixels drawn for a line. This provides the ability to not draw the last pixel of each line segment when the line is to be extended to form a polyline.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	END1										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
0	0	0	0	0	END0										

Bits 15-0 END1

Value = x coordinate (in pixels) of the last pixel to be drawn for the topmost scanline. The first coordinate value is 0. Bits 15-11 are sign bits and must be 0's to indicate a positive value.

Bits 31-16 END0

Value = x coordinate (in pixels) of the first pixel to be drawn for the bottommost scanline. The first coordinate value is 0. Bits 31-27 are sign bits and must be 0's to indicate a positive value.

ViRGE/VX line draw always proceeds from bottom to top. If the requested line is drawn upward with a don't draw the last pixel instruction, the END0 coordinate will be the same as the requested start x coordinate and the END1 coordinate will be 1 less (if drawn from left to right) or 1 more (if drawn from right to left) than the requested end x coordinate. If the requested line is drawn downward, the END1 coordinate will be the same as the requested start x coordinate and the END0 coordinate will be 1 more (if drawn from right to left) or one less (if drawn from left to right) than the requested end x coordinate. See the programming examples for 2D line draw for a more detailed explanation.

Line Draw X Delta Register (LdX) (MMA970)

Read/Write Offset: A970H (2D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
X DELTA HIGH															

Bits 31-0 X DELTA

Value = - (ΔX << 20)/ΔY) with integer division

If the requested line is from coordinates x1,y1 to x2,y2, ΔX is x2 - x1 and ΔY is y2 - y1. (ΔX = x1 - x2 and ΔY = y1 - y2 also works.) The field format is S11.20, i.e, bit 31 is the sign bit (0 = positive), with 11 integer positions and 20 fractional positions.

Line Draw X Start Register (LXSTART) (MMA974)

Read/Write Offset: A974H (2D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
X START HIGH															

Bits 31-0 X START

For an X major line, + X DELTA, value = (x1 << 20) - (X DELTA/2)
For an X major line, - X DELTA, value = (x1 << 20) - (X DELTA/2) + ((1 << 20) - 1)
For a Y major line, value = x1 << 20

For an X major line, the absolute x value increases faster than the absolute y value as the line is drawn. In this case, there may be more than one pixel drawn per scan line. For a Y major line, the absolute y value increases faster than the absolute x value. In this case, at most one pixel will be drawn per scan line. If the requested line is drawn upward, x1 is the requested starting x coordinate. If the requested line is drawn downward, x1 is the requested ending x coordinate. X DELTA is the value programmed in MMA970. The field format is S11.20, i.e, bit 31 is the sign bit (0 = positive), with 11 integer positions and 20 fractional positions.

Line Draw Y Start Register (LYSTART) (MMA978)

Read/Write Offset: A978H (2D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	Y START										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 10-0 Y START

Value = Y coordinate (in scan lines) of first scan line to be drawn

ViRGE/VX draws lines from bottom to top. Therefore this value will be the largest of the requested starting and ending y coordinates.

Bits 31-11 Reserved

Line Draw Y Count Register (LYCNT) (MMA97C)

Read/Write Offset: A97CH (2D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SCAN LINE COUNT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
DIR	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 10-0 SCAN LINE COUNT

Value = [abs (y2 - y1)] +1

y2 is the requested ending y coordinate and y1 is the requested starting y coordinate.

Bits 30-11 Reserved

Bit 31 DIR - Drawing Direction
0 = Draw line from right to left
1 = Draw line from left to right

Polygon Right X Delta Register (PRDX) (MMAD68)

Read/Write Offset: AD68H (Polygon Fill)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RIGHT EDGE X DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RIGHT EDGE X DELTA HIGH															

Bits 31-0 RIGHT EDGE X DELTA

Value = - (ΔX << 20)/ΔY) with integer division

If the requested line is from coordinates x1,y1 to x2,y2, ΔX is x2 - x1 and ΔY is y2 - y1. (ΔX = x1 - x2 and ΔY = y1 - y2 also works.) The field format is S11.20, i.e, bit 31 is the sign bit (0 = positive), with 11 integer positions and 20 fractional positions.

Polygon Right X Start Register (PRXSTART) (MMAD6C)

Read/Write Offset: AD6CH (Polygon Fill)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RIGHT EDGE X START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RIGHT EDGE X START HIGH															

Bits 31-0 RIGHT EDGE X START

For an X major line, value = (x1 << 20) + (RIGHT EDGE X DELTA/2) + (1 << 19)
For a Y major line, value = x1 << 20 + (1 << 19)

For an X major line, the absolute x value increases faster than the absolute y value as the line is drawn. In this case, there may be more than one pixel drawn per scan line. For a Y major line, the absolute y value increases faster than the absolute x value. In this case, at most one pixel will be drawn per scan line. If the requested line is drawn upward, x1 is the requested starting x coordinate. If the requested line is drawn downward, x1 is the requested ending x coordinate. X DELTA is the value programmed in MMA970. The field format is S11.20, i.e, bit 31 is the sign bit (0 = positive), with 11 integer positions and 20 fractional positions.

Polygon Left X Delta Register (PLDX) (MMAD70)

Read/Write Offset: AD70H (Polygon Fill)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LEFT EDGE X DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LEFT EDGE X DELTA HIGH															

Bits 31-0 LEFT EDGE X DELTA

Value = - (ΔX << 20)/ΔY) with integer division

If the requested line is from coordinates x1,y1 to x2,y2, ΔX is x2 - x1 and ΔY is y2 - y1. (ΔX = x1 - x2 and ΔY = y1 - y2 also works.) The field format is S11.20, i.e, bit 31 is the sign bit (0 = positive), with 11 integer positions and 20 fractional positions.

Polygon Left X Start Register (PLXSTART) (MMAD74)

Read/Write Offset: AD74H (Polygon Fill)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LEFT EDGE X START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LEFT EDGE X START HIGH															

Bits 31-0 LEFT EDGE X START

For an X major line, value = (x1 << 20) + (LEFT EDGE X DELTA/2) + (1 << 19)
For a Y major line, value = x1 << 20 + (1 << 19)

For an X major line, the absolute x value increases faster than the absolute y value as the line is drawn. In this case, there may be more than one pixel drawn per scan line. For a Y major line, the absolute y value increases faster than the absolute x value. In this case, at most one pixel will be drawn per scan line. If the requested line is drawn upward, x1 is the requested starting x coordinate. If the requested line is drawn downward, x1 is the requested ending x coordinate. X DELTA is the value programmed in MMA970. The field format is S11.20, i.e, bit 31 is the sign bit (0 = positive), with 11 integer positions and 20 fractional positions.

Polygon Y Start Register (PYSTART) (MMAD78)

Read/Write Offset: AD78H (Polygon Fill)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	Y START										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 10-0 Y START

Value = Y coordinate (in scan lines) of first scan line to be drawn

ViRGE/VX draws lines from bottom to top. Therefore this value will be the largest of the requested starting and ending y coordinates. This value need only be programmed once for each polygon.

Bits 31-11 Reserved

Polygon Y Count Register (PYCNT) (MMAD7C)

Read/Write Offset: AD7CH (Polygon Fill)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SCAN LINE COUNT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	ULE	URE	R	R	R	R	R	R	R	R	R	R	R	R

Bits 10-0 SCAN LINE COUNT

Value = [abs (y2 - y1] +1

The first polygon update proceeds upward to the first vertex. y2 is the requested ending y coordinate for the line leading to that vertex and y1 is the requested starting y coordinate for that line. Both bit 28 and bit 29 will be set to 1 for the first update. For the second polygon update, only the X DELTA for the line extending from the first vertex is re-specified and only the update bit (28 or 29) for that edge is set to 1. The value in this scan line count field is set for the number of scan lines from the first vertex to the second vertex. See the polygon fill programming examples for a more complete explanation of how to program the polygon fill registers at each step to form a complete polygon.

Bits 27-11 Reserved

Bit 28 URE - Update Right Edge
0 = Do not update right edge
1 = Update right edge

Bit 29 ULE - Update Left Edge
0 = Do not update left edge
1 = Update left edge

Bits 31-30 Reserved

19.4 3D REGISTERS

Z-Buffer Base Address Register (Z_BASE) (MMB0D4, MMB4D4)

Read/Write Offset: B0D4H (3D Line), B4D4H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Z-BUFFER BASE ADDRESS													0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	Z-BUFFER BASE ADDRESS					

Bits 2-0 Reserved = 0

Bits 21-3 Z-BUFFER BASE ADDRESS

Value = base address in video memory of the z-buffer used in 3D drawing operations to store depth information for each pixel. Bits 2-0 must be 000b (quadword aligned).

Bits 31-22 Reserved

Destination Base Address Register (DEST_BASE) (MMB0D8, MMB4D8)

Read/Write Offset: B0D8H (3D Line), B4D8H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DESTINATION BASE ADDRESS													0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	DESTINATION BASE ADDRESS					

Bits 2-0 Reserved = 0

Bits 21-3 DESTINATION BASE ADDRESS

Value = base address in video memory of destination data for 2D drawing operations. Bits 2-0 must be 000b (quadword aligned).

This is the 0,0 pixel address in video memory for the screen resolution being used. It will normally be at the start of video memory.

Bits 31-22 Reserved

Left/Right Clipping Register (CLIP_L_R) (MMB0DC, MMB4DC)

Read/Write Offset: B0DCH (3D Line), B4DCH (3D Triangle)
Power-On Default: Undefined

Bit 1 of the Command Set register must be set to 1 for the settings in this register to have effect.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	RIGHT CLIPPING LIMIT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	LEFT CLIPPING LIMIT										

Bits 10-0 RIGHT CLIPPING LIMIT

Value = pixel position of the last pixel to be drawn on each line. The first pixel is 0.

Bits 15-11 Reserved

Bits 26-16 LEFT CLIPPING LIMIT

Value = pixel position of the first pixel to be drawn on each line. The first pixel is 0.

Bits 31-27 Reserved

Top/Bottom Clipping Register (CLIP_T_B) (MMB0E0, MMB4E0)

Read/Write Offset: B0E0H (3D Line), B4E0H (3D Triangle)
Power-On Default: Undefined

Bit 1 of the Command Set register must be set to 1 for the settings in this register to have effect.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	BOTTOM CLIPPING LIMIT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	TOP CLIPPING LIMIT										

Bits 10-0 BOTTOM CLIPPING LIMIT

Value = line position of the last line to be drawn. The first line is 0.

Bits 15-11 Reserved

Bits 26-16 TOP CLIPPING LIMIT

Value = line position of the first line to be drawn. The first line is 0.

Bits 31-27 Reserved

Destination/Source Stride Register (DEST_SRC_STR) (MMB0E4, MMB4E4)

Read/Write Offset: B0E4H (3D Line), B4E4H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	SOURCE STRIDE									0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	DESTINATION STRIDE									0	0	0

Bits 11-0 SOURCE STRIDE (3D Triangle only)

Value = byte offset of vertically adjacent pixels for a flat (not mipmapped) texture map. Bits 2-0 must be 000b.

Bits 15-12 Reserved

Bits 27-16 DESTINATION STRIDE

Value = byte offset of vertically adjacent pixels for the destination data. Bits 2-0 must be 000b.

Bits 31-28 Reserved

Z Stride Register (Z_STRIDE) (MMB0E8, MMB4E8)

Read/Write Offset: B0E8H (3D Line), B4E8H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	Z STRIDE									0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 11-0 Z STRIDE

Value = byte offset of vertically adjacent pixels for the Z-buffer data . Bits 2-0 must be 000b.

Z-buffer data is always 16 bits/pixel. If the destination format is 16 bits/pixel, the Z stride will be the same as the destination stride. Otherwise, the Z stride will differ from the destination stride according to the differing pixel depths.

Bits 31-12 Reserved

Texture Base Address Register (TEX_BASE) (MMB4EC)

Read/Write Offset: B4ECH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEXTURE BASE ADDRESS													0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	TEXTURE BASE ADDRESS					

Bits 2-0 Reserved = 0

Bits 21-3 TEXTURE BASE ADDRESS

Value = base address in video memory of the texture data (flat or mipmapped). Bits 2-0 must be 000b (quadword aligned).

Bits 31-22 Reserved

Texture Border Color Register (TEX_BDR_CLR) (MMB4F0)

Read/Write Offset: B4F0H (3D Triangle)
Power-On Default: Undefined

This is used as the texel color for lighting when texture wrapping is not enabled (bit 26 of the Command Set register is cleared to 0) and the texture rectangle is too small to complete the fill. This must be in the same format at the texture color.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = DAC CLUT index (8 bits/pixel), lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel

Bits 15-8 DATA 2

Value = Reserved (8 bits/pixel), upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

Bit 23-16 DATA 3

Value = Reserved (8, 15 or 16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Fog Color Register (FOG_CLR) (MMB0F4, MMB4F4)

Read/Write Offset: B0F4H (3D Line), B0F4H (3D Triangle)
Power-On Default: Undefined

This is the fog color blended with the pixel color when bit 17 of the Command Set register is set to 1. This operation is also called depth cueing when the fog factor (source alpha) corresponds to the distance from the viewer.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = Lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel

Bits 15-8 DATA 2

Value = Upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

Bit 23-16 DATA 3

Value = Reserved (15 or 16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Color0 Register (COLOR0) (MMB4F8)

Read/Write Offset: B4F8H (3D Triangle)
Power-On Default: Undefined

When using one of the Blend4 modes for texel storage, this register specifies one of the color limits used in the interpolation of the texel color during the generate phase of pixel coloring.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = Lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel

Bits 15-8 DATA 2

Value = Upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

Bit 23-16 DATA 3

Value = Reserved (15/16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Color1 Register (COLOR1) (MMB4FC)

Read/Write Offset: B4FCH (3D Triangle)
Power-On Default: Undefined

When using one of the Blend4 modes for texel storage, this register specifies one of the color limits used in the interpolation of the texel color during the generate phase of pixel coloring.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA 2								DATA 1							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DATA 3							

Bits 7-0 DATA 1

Value = Reserved (8 bits/pixel), lower byte of color data (15/16 bits/pixel), blue color index (24 bits/pixel)

Bits 15-8 DATA 2

Value = Reserved (8 bits/pixel), upper byte of color data (15/16 bits/pixel), green color index (24 bits/pixel)

The 8 bits/pixel color must be programmed to both the DATA 1 and DATA 2 bytes.

Bit 23-16 DATA 3

Value = Reserved (8, 15 or 16 bits/pixel), red color index (24 bits/pixel)

Bits 31-24 Reserved

Command Set Register (CMD_SET) (MMB100, MMB500)

Read/Write Offset: B100H (3D Line), B500H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TB	TEX FLTR MODE			MIPMAP LEVEL SIZE				TEX CLR FORMAT			DEST FORMAT			HC	AE
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
23D	3D COMMAND				TWE	ZB MODE		ZUP	ZB COMP			ABC		FE	TB

Bit 0 AE - Autoexecute
0 = Execute command when this register is written to
1 = Execute command when the highest address register in a drawing type set is written to

The highest address register in a drawing type set is easily seen in Table 20-1, where it is the bottom register in each column. For example, if this bit is set to 1, a 3D line is executed when the 3YCNT (MMB17C) register is written to. Similarly, execution of a 3D Triangle command is based on writing to the TY01_Y12 (MMB57C) register. This setting allows multiple executions of a given command using different parameters without re-writing the Command Set register.

To turn off autoexecute without executing a command, write to this register with this bit cleared to 0 and bits 30-27 programmed to 1111b (NOP).

Bit 1 HC - Hardware Clipping Enable
0 = Hardware clipping disabled
1 = Hardware clipping enabled

The settings in the clipping registers (MMxxDC, MMxxE0) are effective only when this bit is set to 1.

Bits 4-2 DEST FORMAT - Destination Color Format
000 = 8 bits/pixel palettized
001 = 16 bits/pixel (ZRGB1555)
010 = 24 bits/pixel, RGB888

All other values are reserved.

Bits 7-5 TEX CLR FORMAT - Texel Color Format
000 = 32 bits/pixel (ARGB8888)
001 = 16 bits/pixel (ARGB4444)
010 = 16 bits/pixel (ARGB1555)
011 = 8 bits/pixel (Alpha4, Blend4)
100 = 4 bits/pixel (Blend4, low nibble)
101 = 4 bits/pixel (Blend4, high nibble)
110 = 8 bits/pixel (palettized)
111 = YU/YV (16 bits/pixel equivalent)

Bits 11-8 MIPMAP LEVEL SIZE

Value = s, where 2^s is the size of one side of the largest mipmap texture rectangle

For example, a value of 4 specifies the largest mipmap as $2^4 \times 2^4 = 16 \times 16$ texels. The largest allowable s value is 9, which specifies a 512 x 512 texel texture.

Bits 14-12 TEX FLTR MODE - Texture Filtering Mode
000 = M1TPP (MIP_NEAREST)
001 = M2TPP (LINEAR_MIP_NEAREST)
010 = M4TPP (MIP_LINEAR)
011 = M8TPP (LINEAR_MIP_LINEAR)
100 = 1TPP (NEAREST)
101 = V2TPP (used for YU/YV video format - bits 7-5 of this register = 111b)
110 = 4TPP (LINEAR)
111 = Reserved

Only modes with no filtering (000b and 100b) can be used with 8 bits/pixel palettized data. In addition, the texture blending mode must be decal (bits 16-15 of this register = 10b.)

Bits 16-15 TB - Texture Blending Mode
00 = Complex Reflection
01 = Modulate
10 = Decal
11 = Reserved

Bit 17 FE - Fog Enable
0 = Fog color blending disabled
1 = Fog color blending enabled

Fogging is not available for Gouraud shaded triangles or if source alpha is used for blending. If the fog factor (source pixel alpha value) corresponds to the distance from the viewer, this function is also called depth cueing.

Bits 19-18 ABC - Alpha Blending Control
00 = No alpha blending
01 = No alpha blending
10 = Use texture alpha for blending
11 = Use source alpha for blending

Bits 22-20 ZB COMP - Z-buffer Compare Mode
000 = z compare never passes
001 = Pass if Zs > Zzb
010 = Pass if Zs = Zzb
011 = Pass if Zs ≥ Zzb
100 = Pass if Zs < Zzb
101 = Pass if Zs ≠ Zzb
110 = Pass if Zs ≤ Zzb
111 = z compare always passes

Bit 23 ZUP - Z Update Enable
0 = Never update z-buffer
1 = Update z-buffer with new (source) pixel z value if the z compare passes

Bits 25-24 ZB MODE - Z-buffering Mode
00 = Normal Z-buffering
01 = MUX buffering (Z-buffer pass)
10 = MUX buffering (draw buffer pass)
11 = Reserved

Bit 26 TWE - Texture Wrap Enable
0 = Texture wrapping disabled
1 = Texture wrapping enabled

If wrapping is disabled, the texture border color (MMB4F0) may need to be specified.

Bits 30-27 3D COMMAND
0000 = Gouraud Shaded Triangle
0001 = Lit Texture Triangle
0010 = Unlit Texture Triangle
0011 = Reserved
0100 = Reserved
0101 = Lit Texture Triangle with perspective
0110 = Unlit Texture Triangle with perspective
0111 = Reserved
1000 = 3D Line
1001 = Reserved
1010 = Reserved
1011 = Reserved
1100 = Reserved
1101 = Reserved
1110 = Reserved
1111 = NOP

The NOP option is required to turn off autoexecute without executing a command. See the definition for bit 0 of this register.

Bit 31 23D - 2D or 3D Select
0 = A 2D command is being executed
1 = A 3D command is being executed

3D Line Draw GB Delta Register (3dGdY_dBdY) (MMB144)

Read/Write Offset: B144H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLUE DELTA															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GREEN DELTA															

Bits 15-0 BLUE DELTA

Value = Delta value for the accumulation of the blue attribute. The format is S8.7.

Bits 31-16 GREEN DELTA

Value = Delta value for the accumulation of the green attribute. The format is S8.7.

3D Line Draw AR Delta Register (3dAdY_dRdY) (MMB148)

Read/Write Offset: B148H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RED DELTA															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA DELTA															

Bits 15-0 RED DELTA

Value = Delta value for the accumulation of the red attribute. The format is S8.7.

Bits 31-16 ALPHA DELTA

Value = Delta value for the accumulation of the alpha attribute. The format is S8.7.

3D Line Draw GB Start Register (3GS_BS) (MMB14C)

Read/Write Offset: B14CH (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BLUE START														
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
0	GREEN START														

Bits 15-0 BLUE START

Value = Starting value for the accumulation of the blue attribute. The format is S8.7, where S must be 0.

Bits 31-16 GREEN START

Value = Starting value for the accumulation of the green attribute. The format is S8.7, where S must be 0.

3D Line Draw AR Start Register (3AS_RS) (MMB150)

Read/Write Offset: B150H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	RED START														
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
0	ALPHA START														

Bits 15-0 RED START

Value = Starting value for the accumulation of the red attribute. The format is S8.7, where S must be 0.

Bits 31-16 ALPHA START

Value = Starting value for the accumulation of the alpha attribute. The format is S8.7, where S must be 0.

3D Line Draw Z Delta Register (3dZ) (MMB158)

Read/Write Offset: B158H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Z DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Z DELTA HIGH															

Bits 31-0 Z DELTA

Value = Delta value for the accumulation of the Z attribute. The format is S16.15.

3D Line Draw Z Start Register (3ZSTART) (MMB15C)

Read/Write Offset: B15CH (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Z START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	Z START HIGH														

Bits 31-0 Z START

Value = Starting value for the accumulation of the Z attribute. The format is S16.15, where S must be 0.

3D Line Draw Endpoints Register (3XEND0_END1) (MMB16C)

Read/Write Offset: B16CH (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	END1										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
0	0	0	0	0	END0										

Bits 15-0 END1

Value = x coordinate (in pixels) of the last pixel to be drawn for the topmost scanline. The first coordinate value is 0. Bits 15-11 are sign bits and must be 0's to indicate a positive value.

Bits 31-16 END0

Value = x coordinate (in pixels) of the first pixel to be drawn for the bottommost scanline. The first coordinate value is 0. Bits 31-27 are sign bits and must be 0's to indicate a positive value.

3D Line Draw X Delta Register (3dX) (MMB170)

Read/Write Offset: B170H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X DELTA LOW															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
X DELTA HIGH															

Bits 31-0 X DELTA

Value = Delta value for the accumulation of the X attribute. The format is S11.20.

3D Line Draw X Start Register (3XSTART) (MMB174)

Read/Write Offset: B174H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X START LOW															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
0	X START HIGH														

Bits 31-0 X START

Value = Starting value for the accumulation of the X attribute. The format is S11.20, where S must be 0.

3D Line Draw Y Start Register (3YSTART) (MMB178)

Read/Write Offset: B178H (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	Y START										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 10-0 Y START

Value = Y coordinate (in scan lines) of first scan line to be drawn

ViRGE/VX draws lines from bottom to top. Therefore this value will be the largest of the requested starting and ending y coordinates.

Bits 31-11 Reserved

3D Line Draw Y Count Register (3YCNT) (MMB17C)

Read/Write Offset: AB1CH (3D Line)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SCAN LINE COUNT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
DIR	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 10-0 SCAN LINE COUNT

Value = The number of scan lines to be rendered

Bits 30-11 Reserved

Bit 31 DIR - Drawing Direction
0 = Draw line from right to left
1 = Draw line from left to right

Triangle Base V Register (TBV) (MMB504)

Read/Write Offset: B504H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASE V															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	BASE V			

Bits 19-0 BASE V

Value = Base vertical coordinate value for texels. The format is 12.8.

This is the common offset for all V coordinate values for textures.

Bits 31-20 Reserved

Triangle Base U Register (TBU) (MMB508)

Read/Write Offset: B508H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASE U															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	BASE U			

Bits 19-0 BASE U

Value = Base horizontal coordinate value for texels. The format is 12.8.

This is the common offset for all U coordinate values for textures.

Bits 31-20 Reserved

Triangle WX Delta Register (TdWdX) (MMB50C)

Read/Write Offset: B50CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WX DELTA LOW															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
WX DELTA HIGH															

Bits 31-0 WX DELTA

Value = Delta value for the accumulation of the W attribute (homogeneous coordinate) with respect to X. The format is S12.19.

W is the depth coordinate for 3D texture maps.

Triangle WY Delta Register (TdWdY) (MMB510)

Read/Write Offset: B510H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WY DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WY DELTA HIGH															

Bits 31-0 WY DELTA

Value = Delta value for the accumulation of the W attribute (homogeneous coordinate) with respect to Y. The format is S12.19.

W is the depth coordinate for 3D texture maps.

Triangle W Start Register (TWS) (MMB514)

Read/Write Offset: B514H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
W START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	W START HIGH														

Bits 31-0 W START

Value = Starting value for the accumulation of the W attribute (homogeneous coordinate). The format is S12.19, where S must be 0.

W is the depth coordinate for 3D texture maps.

Triangle DX Delta Register (TdDdX) (MMB518)

Read/Write Offset: B518H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DX DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DX DELTA HIGH															

Bits 31-0 DX DELTA

Value = Delta value for the accumulation of the D attribute with respect to X. The format is S4.8.19 (1 sign bit, 4 wrap bits, 8 integer bits, 19 fractional bits - the wrap bits specify the number of map edge wrap arounds allowed for the texture.)

Wrapping is enabled by setting bit 26 of the Command Set register to 1. The D attribute specifies the level within a texture mipmap.

Triangle VX Delta Register (TdVdX) (MMB51C)

Read/Write Offset: B51CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VX DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VX DELTA HIGH															

Bits 31-0 VX DELTA

Value = Delta value for the accumulation of the V attribute with respect to X. The format is S24.7 if perspective is enabled (3D command = 0101b or 0110b). The format is S12.8.11 without perspective enabled. This format is 1 sign bit, 12 integer bits, 8 filter bits and 11 fractional bits.

Triangle UX Delta Register (TdUdX) (MMB520)

Read/Write Offset: B520H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UX DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UX DELTA HIGH															

Bits 31-0 UX DELTA

Value = Delta value for the accumulation of the U attribute with respect to X. The format is S24.7 if perspective is enabled (3D command = 0101b or 0110b). The format is S12.8.11 without perspective enabled. This format is 1 sign bit, 12 integer bits, 8 filter bits and 11 fractional bits.

Triangle DY Delta Register (TdDdY) (MMB524)

Read/Write Offset: B524H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DY DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DY DELTA HIGH															

Bits 31-0 DY DELTA

Value = Delta value for the accumulation of the D attribute with respect to Y. The format is S4.8.19 (1 sign bit, 4 wrap bits, 8 integer bits, 19 fractional bits - the wrap bits specify the number of map edge wrap arounds allowed for the texture.)

Wrapping is enabled by setting bit 26 of the Command Set register to 1. The D attribute specifies the level within a texture mipmap.

Triangle VY Delta Register (TdVdY) (MMB528)

Read/Write Offset: B528H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VY DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VY DELTA HIGH															

Bits 31-0 VY DELTA

Value = Delta value for the accumulation of the V attribute with respect to Y. The format is S24.7 if perspective is enabled (3D command = 0101b or 0110b). The format is S12.8.11 without perspective enabled. This format is 1 sign bit, 12 integer bits, 8 filter bits and 11 fractional bits.

Triangle UY Delta Register (TdUdY) (MMB52C)

Read/Write Offset: B52CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UY DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UY DELTA HIGH															

Bits 31-0 UY DELTA

Value = Delta value for the accumulation of the U attribute with respect to Y. The format is S24.7 if perspective is enabled (3D command = 0101b or 0110b). The format is S12.8.11 without perspective enabled. This format is 1 sign bit, 12 integer bits, 8 filter bits and 11 fractional bits.

Triangle D Start Register (TDS) (MMB530)

Read/Write Offset: B530H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D START															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	D START														

Bits 31-0 D START

Value = Starting value for the accumulation of the D attribute. The format is S4.8.19 (1 sign bit = 0, 4 wrap bits, 8 integer bits, 19 fractional bits - the wrap bits specify the number of map edge wrap arounds allowed for the texture.)

Wrapping is enabled by setting bit 26 of the Command Set register to 1. The D attribute specifies the level within a texture mipmap.

Triangle V Start Register (TVS) (MMB534)

Read/Write Offset: B534H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	V START HIGH														

Bits 31-0 V START

Value = Starting value for the accumulation of the V attribute. The format is S24.7 if perspective is enabled (3D command = 0101b or 0110b). The format is S12.8.11 without perspective enabled. This format is 1 sign bit, 12 integer bits, 8 filter bits and 11 fractional bits. In either case, the sign bit must be 0.

The V attribute is the vertical coordinate value for a texel.

Triangle U Start Register (TUS) (MMB538)

Read/Write Offset: B538H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	U START HIGH														

Bits 31-0 U START

Value = Starting value for the accumulation of the U attribute. The format is S24.7 if perspective is enabled (3D command = 0101b or 0110b). The format is S12.8.11 without perspective enabled. This format is 1 sign bit, 12 integer bits, 8 filter bits and 11 fractional bits. In either case, the sign bit must be 0.

The U attribute is the horizontal coordinate value for a texel.

Triangle GBX Delta Register (TdGdX_dBdX) (MMB53C)

Read/Write Offset: B53CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLUE X DELTA															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GREEN X DELTA															

Bits 15-0 BLUE X DELTA

Value = Delta value for the accumulation of the blue attribute with respect to X. The format is S8.7.

Bits 31-16 GREEN X DELTA

Value = Delta value for the accumulation of the green attribute with respect to X. The format is S8.7.

Triangle ARX Delta Register (TdAdX_dRdX) (MMB540)

Read/Write Offset: B540H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RED X DELTA															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA X DELTA															

Bits 15-0 RED X DELTA

Value = Delta value for the accumulation of the red attribute with respect to X. The format is S8.7.

Bits 31-16 ALPHA X DELTA

Value = Delta value for the accumulation of the alpha attribute with respect to X. The format is S8.7.

Triangle GBY Delta Register (TdGdY_dBdY) (MMB544)

Read/Write Offset: B544H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLUE Y DELTA															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GREEN Y DELTA															

Bits 15-0 BLUE Y DELTA

Value = Delta value for the accumulation of the blue attribute with respect to Y. The format is S8.7.

Bits 31-16 GREEN Y DELTA

Value = Delta value for the accumulation of the green attribute with respect to Y. The format is S8.7.

Triangle ARY Delta Register (TdAdY_dRdY) (MMB548)

Read/Write Offset: B548H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RED Y DELTA															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA Y DELTA															

Bits 15-0 RED Y DELTA

Value = Delta value for the accumulation of the red attribute with respect to Y. The format is S8.7.

Bits 31-16 ALPHA Y DELTA

Value = Delta value for the accumulation of the alpha attribute with respect to Y. The format is S8.7.

Triangle GB Start Register (TGS_BS) (MMB54C)

Read/Write Offset: B54CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	BLUE START														
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	GREEN START														

Bits 15-0 BLUE START

Value = Starting value for the accumulation of the blue attribute. The format is S8.7, where S must be 0.

Bits 31-16 GREEN START

Value = Starting value for the accumulation of the green attribute. The format is S8.7, where S must be 0.

Triangle AR Start Register (TAS_RS) (MMB550)

Read/Write Offset: B550H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	RED START														
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
0	ALPHA START														

Bits 15-0 RED START

Value = Starting value for the accumulation of the red attribute. The format is S8.7, where S must be 0.

Bits 31-16 ALPHA START

Value = Starting value for the accumulation of the alpha attribute. The format is S8.7, where S must be 0.

Triangle ZX Delta Register (TdZdX) (MMB554)

Read/Write Offset: B554H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ZX DELTA LOW															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
ZX DELTA HIGH															

Bits 31-0 ZX DELTA

Value = Delta value for the accumulation of the Z attribute with respect to X. The format is S16.15.

Triangle ZY Delta Register (TdZdY) (MMB558)

Read/Write Offset: B558H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ZY DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ZY DELTA HIGH															

Bits 31-0 ZY DELTA

Value = Delta value for the accumulation of the Z attribute with respect to Y. The format is S16.15.

Triangle Z Start Register (TZS) (MMB55C)

Read/Write Offset: B55CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Z START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	Z START HIGH														

Bits 31-0 Z START

Value = Starting value for the accumulation of the Z attribute. The format is S16.15, where S must be 0.

The Z attribute is used in conjunction with z-buffering.

Triangle XY12 Delta Register (TdXdY12) (MMB560)

Read/Write Offset: B560H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XY12 DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
XY12 DELTA HIGH															

Bits 31-0 XY12 DELTA

Value = Delta value for the accumulation of the X attribute with respect to Y along the 12 side. The format is S11.20.

See 3D Programming in Section 15 for an explanation of this field.

Triangle X12 End Register (TXEND12) (MMB564)

Read/Write Offset: B564H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X12 END LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	X12 END HIGH														

Bits 31-0 X12 END

Value = X coordinate for the last pixel drawn for side 12. The format is S11.20, where S must be 0.

See 3D Programming in Section 15 for an explanation of this field.

Triangle XY01 Delta Register (TdXdY01) (MMB568)

Read/Write Offset: B568H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XY01 DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
XY01 DELTA HIGH															

Bits 31-0 XY01 DELTA

Value = Delta value for the accumulation of the X attribute with respect to Y along the 01 side. The format is S11.20.

See 3D Programming in Section 15 for an explanation of this field.

Triangle X01 End Register (TXEND01) (MMB56C)

Read/Write Offset: B56CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X01 END LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	X01 END HIGH														

Bits 31-0 X01 END

Value = X coordinate for the last pixel drawn for side 01. The format is S11.20, where S must be 0.

See 3D Programming in Section 15 for an explanation of this field.

Triangle XY02 Delta Register (TdXdY02) (MMB570)

Read/Write Offset: B570H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XY02 DELTA LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
XY02 DELTA HIGH															

Bits 31-0 XY02 DELTA

Value = Delta value for the accumulation of the X attribute with respect to Y along the 02 side. The format is S11.20.

See 3D Programming in Section 15 for an explanation of this field.

Triangle X Start Register (TXS) (MMB574)

Read/Write Offset: B574H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	X START HIGH														

Bits 31-0 X START

Value = Starting value for the accumulation of the X attribute. The format is S11.20, where S must be 0.

Triangle Y Start Register (TYS) (MMB578)

Read/Write Offset: B578H (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Y START LOW															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	Y START HIGH														

Bits 31-0 Y START

Value = Starting value for the accumulation of the Y attribute. The format is S11.20, where S must be 0.

Triangle Y Count Register (TY01_Y12) (MMB57C)

Read/Write Offset: B57CH (3D Triangle)
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SCAN LINE COUNT 12										
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
L/R	R	R	R	R	SCAN LINE COUNT 01										

Bits 10-0 SCAN LINE COUNT 12

Value = The number of scan lines required to render the 12 side of the triangle.

See 3D Programming in Section 15 for a graphic description of this field. Either this field or the SCAN LINE COUNT 01 field below must be non-zero for the S3d Engine to draw a triangle.

Bits 15-11 Reserved

Bits 26-16 SCAN LINE COUNT 01

Value = The number of scan lines required to render the 01 side of the triangle.

See 3D Programming in Section 15 for a graphic description of this field. Either this field or the SCAN LINE COUNT12 field above must be non-zero for the S3d Engine to draw a triangle.

Bits 30-27 Reserved

Bit 31 L/R - Left/Right Drawing Direction
0 = Render the triangle from right to left
1 = Render the triangle from left to right

The triangle must always be rendered in the direction starting with the triangle side with the largest Y component. See 3D Programming in Section 15 for a graphic description.

Section 20: Streams Processor Register Descriptions

This section describes the Streams Processor registers.

In all register bit descriptions, the letter "R" identifies reserved bits (a reserved bit's read value is undefined unless noted, and you may write only zero to a reserved bit).

Primary Stream Control (MM8180)

Read/Write Address: 8180H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	PSFC			R	PSIDF			R	R	R	R	R	R	R	R

Bits 23-0 Reserved

Bits 26-24 PSIDF - Primary Stream Input Data Format
000 = RGB-8 (CLUT)
001 = Reserved
010 = Reserved
011 = Reserved
100 = Reserved
101 = RGB-16 (5.6.5)
110 = RGB-24 (8.8.8)
111 = XRGB-32 (X.8.8.8)

Bit 27 Reserved

Bits 30-28 PSFC - Primary Stream Filter Characteristics
000 = Primary stream
001 = Primary stream for 2X stretch (replication)
010 = Primary stream, bi-linear for 2X stretch (interpolation)
Other values reserved

Bit 31 Reserved

Color/Chroma Key Control (MM8184)

Read/Write Adds: 8184H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G/U/Cb KEY (LOW)								B/V/Cr KEY (LOW)							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	CKE	R	RGB CC			R/Y KEY (LOW)							

Bits 7-0 B/V/Cr key value (lower bound for chroma)

Bits 15-8 G/U/Cb key value (lower bound for chroma)

Bits 23-16 R/Y key value (lower bound for chroma)

Bits 26-24 RGB CC - RGB Color Comparison Precision
000 = Compare bit 7 of RGB (compare red bit 7's, green bit 7's and blue bit 7's)
001 = Compare bits 7-6 of RGB
010 = Compare bits 7-5 of RGB
011 = Compare bits 7-4 of RGB
100 = Compare bits 7-3 of RGB
101 = Compare bits 7-2 of RGB
110 = Compare bits 7-1 of RGB
111 = Compare bits 7-0 of RGB

Bit 27 Reserved

Bit 28 CKE - Color Key Enable
0 = Disable color or chroma keying
1 = Enable color or chroma keying

Bits 31-29 Reserved

Secondary Stream Control (MM8190)

Read/Write Address: 8190H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	DDA HORIZONTAL ACCUMULATOR												
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	SFC			R	SDIF			R	R	R	R	R	R	R	R

Bits 12-0 DDA Horizontal Accumulator Initial Value

Value = 2 (W0-1) - (W1-1), where W0 is the line width in pixels before scaling and W1 is the line width in pixels after scaling. This is a signed value.

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 23-13 Reserved

Bits 26-24 SDIF - Secondary Stream Input Data Format
000 = Reserved
001 = YCbCr-16 (4.2.2), 16-240 input range
010 = YUV-16 (4.2.2), 0-255 input range
011 = Reserved
100 = YUV (2.1.1)
101 = RGB-16 (5.6.5)
110 = RGB-24 (8.8.8)
111 = XRGB-32 (X.8.8.8

When this field is programmed, the value does not take effect until the next VSYNC.

Bit 27 Reserved

Bits 30-28 SFC - Secondary Stream Filter Characteristics
000 = Secondary stream
001 = Secondary stream, linear, 0-2-4-2-0, for X stretch
010 = Secondary stream, bi-linear, for 2X to 4X stretch
011 = Secondary stream, linear, 1-2-2-2-1, for 4X stretch
Other values reserved

When this field is programmed, the value does not take effect until the next VSYNC.

Bit 31 Reserved

Chroma Key Upper Bound (MM8194)

Read/Write Address: 8194H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U/Cb KEY (UPPER)								V/Cr KEY (UPPER)							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	Y KEY (UPPER)							

Bits 7-0 V/Cr key value (upper bound)

Bits 15-8 U/Cb key value (upper bound)

Bits 23-16 Y key value (upper bound)

Bits 31-24 Reserved

Secondary Stream Stretch/Filter Constants (MM8198)

Read/Write Address: 8198H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	K2 HORIZONTAL SCALE FACTOR											
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	K1 HORIZONTAL SCALE FACTOR											

Bits 11-0 K1 Horizontal Scale Factor

Value = W0-1, where W0 is the width in pixels of the initial output window (before scaling)

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 15-12 Reserved

Bits 27-16 K2 Horizontal Scale Factor

Value = W0-W1, where W0 is the initial (unscaled) window width in pixels and W1 is the final output window width in pixels. This is a signed value and will always be negative.

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 31-28 Reserved

Blend Control (MM81A0)

Read/Write Address: 81A0H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	KP			R	R	R	R	R	KS			R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	COMP MODE			R	R	R	R	R	R	R	R

Bits 1-0 Reserved

Bits 4-2 Ks

Value = secondary stream blend coefficient

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 9-5 Reserved

Bits 12-10 Kp

Value = primary stream blend coefficient

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 23-13 Reserved

Bits 26-24 Compose Mode
000 = Secondary stream opaque overlay on primary stream
001 = Primary stream opaque overlay on secondary stream
010 = Dissolve, [Pp x Kp + Ps x (8 - Kp)]/8, ignore Ks
011 = Fade, [Pp x Kp + Ps x Ks]/8, where Kp + Ks must be ≤ 8
100 = Reserved
101 = Color key on primary stream (secondary stream overlay on primary stream)
110 = Color or chroma key on secondary stream (primary stream overlay on secondary stream)
111 = Reserved

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 31-27 Reserved

Primary Stream Frame Buffer Address 0 (MM81C0)

Read/Write Address: 81C0H
Power-on Default: Undefined

If a primary stream is enabled, this register specifies the starting address in the frame buffer.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIMARY BUFFER ADDRESS 0															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	PRIMARY BUFFER ADDRESS 0					

Bits 21-0 Value = Primary stream frame buffer starting address 0

This value must be quadword aligned.

Bits 31-22 Reserved

Primary Stream Frame Buffer Address 1 (MM81C4)

Read/Write Address: 81C4H
Power-on Default: Undefined

If the primary stream is double buffered, this register specifies the starting address in the frame buffer for the second buffer.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIMARY BUFFER ADDRESS 1															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	PRIMARY BUFFER ADDRESS 1					

Bits 21-0 Value = Primary stream frame buffer starting address 1

This value must be quadword aligned.

Bits 31-22 Reserved

Primary Stream Stride (MM81C8)

Read/Write Address: 81C8H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	PRIMARY STREAM STRIDE											
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 11-0 Primary stream stride

Value = byte offset of vertically adjacent pixels in the primary stream buffer(s)

If double buffering is used, the stride must be the same for both buffers.

Bits 31-12 Reserved

Double Buffer/LPB Support (MM81CC)

Read/Write Address: 81CCH
Power-on Default: xxxxxx00H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	LST	LSL	LIS	R	SBS		PBS
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 PBS - Primary Stream Buffer Select
0 = Primary frame buffer starting address 0 (MM81C0_21-0) used for the primary stream
1 = Primary frame buffer starting address 1 (MM81C4_21-0) used for the primary stream

Bits 2-1 SBS - Secondary Stream Buffer Select
00 = Secondary frame buffer starting address 0 (MM81D0_21-0) used for the secondary stream
01 = Secondary frame buffer starting address 1 (MM81D4_21-0) used for the secondary stream
10 = Secondary frame buffer starting address 0 (MM81D0_21-0) used for the secondary stream and LPB frame buffer starting address 0 (MMFF0C_21-0) used for the LPB input stream OR secondary frame buffer starting address 1 (MM81D4_21-0) used for the secondary stream and LPB frame buffer starting address 1 (MMFF10_21-0) used for the LPB input stream. Which alternative applies is determined by LPB starting address register selected by bit 4 of this register
11 = Secondary frame buffer starting address 0 (MM81D0_21-0) used for the secondary stream and LPB frame buffer starting address 1 (MMFF10_21-0) used for the LPB input stream OR secondary frame buffer starting address 1 (MM81D4_21-0) used for the secondary stream and LPB frame buffer starting address 0 (MMFF0C_21-0) used for the LPB input stream. Which alternative applies is determined by the LPB starting address register selected by bit 4 of this register

Bit 3 Reserved

Bit 4 LIS - LPB Input Buffer Select
0 = LPB frame buffer starting address 0 (MMFF0C_21-0) used for the LPB input
1 = LPB frame buffer starting address 1 (MMFF10_21-0) used for the LPB input

This bit selects the starting address for writing LPB data into the frame buffer. When the value programmed to this bit takes effect is determined by the setting of bit 5 of this register. This bit can be toggled at the completion of writing all the data for a frame to the frame buffer via bit 6 of this register

Bit 5 LSL - LPB Input Buffer Select Loading
0 = The value programmed into bit 4 of this register takes effect immediately
1 = The value programmed into bit 4 of this register takes effect at the next end of frame (completion of writing all the data for a frame into the frame buffer)

Bit 6 LST - LPB Input Buffer Select Toggle
0 = End of frame (completion of writing all the data for a frame into the frame buffer) has no effect on the setting of bit 4 of this register
1 = End of frame causes the setting of bit 4 of this register to toggle

Bits 31-7 Reserved

Secondary Stream Frame Buffer Address 0 (MM81D0)

Read/Write Address: 81D0H
Power-on Default: Undefined

If a secondary stream is enabled, this register specifies the starting address in the frame buffer

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SECONDARY BUFFER ADDRESS 0															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	SECONDARY BUFFER ADDRESS 0					

Bits 21-0 Value = Secondary stream frame buffer starting address 0

This value must be quadword aligned.

Bits 31-22 Reserved

Secondary Stream Frame Buffer Address 1 (MM81D4)

Read/Write Address: 81D4H
Power-on Default: Undefined

If the secondary stream is double buffered, this register specifies the starting address in the frame buffer for the second buffer.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SECONDARY BUFFER ADDRESS 1															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	SECONDARY BUFFER ADDRESS 1					

Bits 21-0 Value = Secondary stream frame buffer starting address 1

This value must be quadword aligned.

Bits 31-22 Reserved

Secondary Stream Stride (MM81D8)

Read/Write Address: 81D8H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	SECONDARY STREAM STRIDE											
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 11-0 Secondary stream stride

Value = byte offset of vertically adjacent pixels in the secondary stream buffer(s)

If double buffering is used, the stride must be the same for both buffers.

Bits 31-12 Reserved

Opaque Overlay Control (MM81DC)

Read/Write Address: 81DCH
Power-on Default: Undefined except bits 31-30 are 00b.

When an opaque overlay mode is being used (bits 26-24 of MM81A0 = 000b or 001b), the fields in this register can be programmed to eliminate the fetching of the pixels for the rectangular area under the top (opaque) window. This reduces the memory bandwidth requirements. The bottom window should be full-screen when this feature is enabled. None of the fields in this register have an effect unless bit 31 is set to 1. Note that only horizontal coordinates must be specified. The vertical coordinates are handled automatically by the hardware.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	PIXEL STOP FETCH										R	R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
OOC	TSS	R	PIXEL RESUME FETCH										R	R	R

Bits 2-0 Reserved

Bits 12-3 Pixel Stop Fetch

Value = [Offset in quadwords from the background starting pixel horizontal position to the first pixel of the line not to be fetched from memory (hidden background)] + 1 quadword

If the primary stream is the background, MM81F0_26-16 define the starting position for each line in the background window (X0) and MM81F8_26-16 define the first pixel position for each line in the top window (X1). The latter is the first background pixel that does not need to be fetched. The value programmed in this field is then [(X1 -

X0) x bytes per pixel/8] +1. If the result is a fraction, it is rounded up the next highest integer. This gives the required quadword offset (O) for this field. This value is also used in the calculation for the field value of bits 28-19 of this register.

If the secondary stream is the background, the value is [(X0 - X1) x bytes per pixel/8] + 1.

Bits 18-13 Reserved

Bits 28-19 Pixel Resume Fetch

Value = {Offset in quadwords from the background starting pixel horizontal position to the line position of the resumption of pixel fetching from memory (i.e., visible background)} - 1 quadword

The value is determined by adding the Pixel Stop Fetch field value (O) above (bits 12-3) to the width in quadwords of the top window (W). The width of the top window in pixels (P) is found in MM81F4_26-16 if the primary stream is on top and in MM81FC_26-16 if the secondary stream is on top. W in quadwords = P x bytes per pixel/8. If this is a fraction, the result is truncated to the next lowest integer. The value in this field is then [W + O] - 1.

Bit 29 Reserved

Bit 30 TSS - Top Stream Select
0 = Secondary stream on top
1 = Primary stream on top

Bit 31 OOC - Opaque Overlay Control Enable
0 = Opaque overlay control disabled
1 = Opaque overlay control enabled

K1 Vertical Scale Factor (MM81E0)

Read/Write Address: 81E0H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	K1 VERTICAL SCALE FACTOR											
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 11-0 K1 Vertical Scale Factor

Value = [height (in lines) of the initial output window (before scaling)] - 1

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 31-12 Reserved

K2 Vertical Scale Factor (MM81E4)

Read/Write Address: 81E4H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	K2 VERTICAL SCALE FACTOR												
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 12-0 K2 Vertical Scale Factor

Value = [height (in lines) of the initial output window (before scaling)] - [height (in lines) of the final output window (after scaling)]

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 31-13 Reserved

DDA Vertical Accumulator Initial Value (MM81E8)

Read/Write Address: 81E8H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EVF	R	R	DDA VERTICAL ACCUMULATOR												
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 12-0 DDA Vertical Accumulator Initial Value

Value = 2's complement of [height (in lines) of the output window after scaling] - 1

When this field is programmed, the value does not take effect until the next VSYNC.

Bits 14-13 Reserved

Bit 15 EVF - Enable Vertical Filtering
0 = Line replication used
1 = Vertical filtering (interpolation) enabled

Vertical filtering is only available when CR67_3-2 = 11b.

Bits 31-12 Reserved

Streams FIFO Control (MM81EC)

Read/Write Address: 81ECH
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	TSM	EWS	R	RP

Bits 17-0 Reserved

Bit 18 EWS - EDO Memory Wait State Control (LPB Memory Cycles Only)
0 = Standard 2-cycle memory operation
1 = 1-cycle EDO memory operation

Bit 19 TSM - Tri-state Memory Data Lines
0 = Tri-state PD[63:16] during ROM cycles
1 = Do not tri-state PD[63:16] during ROM cycles

PD[15:0] are driven with the ROM address. This bit should normally be left at its default value of 0.

Bits 31-20 Reserved

Primary Stream Window Start Coordinates (MM81F0)

Read/Write Address: 81F0H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	PRIMARY STREAM Y-START										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	PRIMARY STREAM X-START										

Bits 10-0 Primary Stream Y-Start

Value = Screen line number +1 of the first line of the primary stream window

Bits 15-11 Reserved

Bits 26-16 Primary Stream X-Start

Value = Screen pixel number +1 of the first pixel of the primary stream window

Bits 31-27 Reserved

Primary Stream Window Size (MM81F4)

Read/Write Address: 81F4H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	PRIMARY STREAM HEIGHT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	PRIMARY STREAM WIDTH										

Bits 10-0 Primary Stream Height

Value = Number of lines displayed in the primary stream window

Bits 15-11 Reserved

Bits 26-16 Primary Stream Width

Value = Number of pixels -1 displayed in each line in the primary stream window

Bits 31-27 Reserved

Secondary Stream Window Start Coordinates (MM81F8)

Read/Write Address: 81F8H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SECONDARY STREAM Y-START										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	SECONDARY STREAM X-START										

Bits 10-0 Secondary Stream Y-Start

Value = Screen line number +1 of the first line of the secondary stream window

Bits 15-11 Reserved

Bits 26-16 Secondary Stream X-Start

Value = Screen pixel number +1 of the first pixel of the secondary stream window

Bits 31-27 Reserved

Secondary Stream Window Size (MM81FC)

Read/Write Address: 81FCH
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	SECONDARY STREAM HEIGHT										
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	SECONDARY STREAM WIDTH										

Bits 10-0 Secondary Stream Height

Value = Number of lines displayed in the secondary stream window

Bits 15-11 Reserved

Bits 26-16 Secondary Stream Width

Value = Number of pixels -1 displayed in each line in the primary stream window

Bits 31-27 Reserved

Section 21: Memory Port Controller Register Descriptions

This section describes the Memory Port Controller (MPC) Registers for ViRGE/VX. These registers are used to adjust memory control signals and control the video data FIFOs.

In all register bit descriptions, the letter "R" identifies reserved bits (a reserved bit's read value is undefined unless noted, and you may write only zero to a reserved bit).

FIFO Control (MM8200)

Read/Write Offset:8200H
Power-On Default: 00010400H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PS THRESHOLD				R	SS THRESHOLD					R	P/S BOUNDARY				
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	PS

Bits 4-0 P/S BOUNDARY - Primary/Secondary Stream FIFO Boundary
00000 = Primary Stream = 24 slots; Secondary stream = 0 slots
01000 = Primary Stream = 16 slots; Secondary stream = 8 slots
01100 = Primary Stream = 12 slots; Secondary stream = 12 slots
10000 = Primary Stream = 8 slots; Secondary stream = 16 slots
11000 = Primary Stream = 0 slots; Secondary stream = 24 slots

All other values are reserved and must not be programmed. Each slot holds 1 quad-word.

Note: The above definition strictly applies only for CR67_3-2 = 01b (secondary stream over a VGA primary stream). For full streams operation (CR67_3-2 = 11b) and vertical filtering disabled, this field should be programmed to 11000b to allot 24 slots to the secondary stream. For full streams operation and vertical filtering enabled (MM81E8_15 = 1), this field should be programmed to 01100b to allot 12 slots to each stream.

Bit 5 Reserved

Bits 10-6 SS THRESHOLD - Secondary Stream Threshold

Value = Number of secondary stream FIFO slots

When the secondary stream FIFO empties down to this value, an internal signal is generated requesting refilling of the secondary stream FIFO. This value must be ≤ the secondary stream FIFO size specified in bits 4-0 of this register.

Bit 11 Reserved

Bits 16-12 PS THRESHOLD - Primary Stream Threshold

Value = Number of primary stream FIFO slots

When the primary stream FIFO empties down to this value, an internal signal is generated requesting refilling of the primary stream FIFO. This value must be ≤ the primary stream FIFO size specified in bits 4-0 of this register.

Bits 31-17 Reserved

MIU Control Register (MM8204)

Read/Write Offset: 8204H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	RL	RP	RI
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 RI -$\overline{\text{RAS}}$ Inactive
0 = $\overline{\text{RAS}}$ stays low until the start of the next memory cycle
1 = $\overline{\text{RAS}}$ goes high at the end of the memory cycle

Bit 1 RP - RAS Pre-charge Control
0 = RAS pre-charge specified by CR68_3 (2.5 or 3.5 MCLKs)
1 = RAS pre-charge = 1.5 MCLKs

Bit 2 RL - RAS Low
0 = RAS low specified by CR68_2 (3.5 or 4.5 MCLKs)
1 = RAS low = 2.5 MCLKs

Bits 6-3 Reserved

Bit 7 COA - CAS/OE Adjust
0 = CAS/OE active trailing edge delay specified by CR68_1-0
1 = CAS/OE active pulse delay specified by CR68_1-0

Bit 8 WED - WE Delay
0 = WE active trailing edge delay specified by CR6F_4-3
1 = WE active pulse delay specified by CR6F_4-3

Bit 9 LPR - Low Priority Request Control
0 = A low priority primary or secondary stream memory request is service when an S3d Engine request goes inactive
1 = A low priority primary or secondary stream memory request is serviced only after an S3d Engine request has been inactive for more than 3 MCLKs.

Bit 10 DRR - Delay RAS Rising Edge
0 = No delay of RAS rising edge
1 = Delay RAS rising edge by 1/2 MCLK

Bits 31-11 Reserved

Streams Timeout Register (MM8208)

Read/Write Offset: 8208H
Power-On Default: 00000808H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PS TIMEOUT								SS TIMEOUT							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	PST

Bits 7-0 SS TIMEOUT - Secondary Stream Timeout

Value = number of MCLKs that the secondary stream is given read access to video memory before its grant is withdrawn

Bits 15-8 PS TIMEOUT - Primary Stream Timeout

Value = number of MCLKs that the primary stream is given read access to video memory before its grant is withdrawn

Bit 16 PST - Primary/Secondary Tiebreaker
0 = Primary wins in case of a tie
1 = Secondary wins in case of a tie

This bit is effective when the primary and secondary streams have simultaneous requests for video memory access pending.

Bits 31-17 Reserved

Miscellaneous Timeout Register (MM820C)

Read/Write Offset: 820CH
Power-On Default: 08080810H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S3D ENGINE TIMEOUT								CPU TIMEOUT							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
EXT TIMEOUT								LPB TIMEOUT							

Bits 7-0 CPU TIMEOUT

Value = number of MCLKs that the CPU is given access to video memory before its grant is withdrawn

Bits 15-8 S3D ENGINE TIMEOUT

Value = number of MCLKs that the S3D Engine is given access to video memory before its grant is withdrawn

Bits 23-16 LPB TIMEOUT

Value = number of MCLKs that the LPB is given write access to video memory before its grant is withdrawn

Bits 31-24 EXT TIMEOUT

Value = number of MCLKs that another memory master is given access to video memory before its grant is withdrawn

DMA Read Base Address Register (MM8220)

Read/Write Offset: 8220H
Power-On Default: Undefined

This register is used when the CPU is doing DMA transfers from video memory as specified by clearing bit 1 of MM8580 to 0 (read) and setting bit 0 of MM8588 to 1 (video DMA enable).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMA READ BASE ADDRESS													0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	DMA READ BASE ADDRESS						

Bits 2-0 Reserved = 0

Bits 22-3 DMA READ BASE ADDRESS

Value = Starting address in video memory for data to be DMAed to system memory (quadword aligned)

Bits 31-23 Reserved

DMA Read Stride/Width Register (MM8224)

Read/Write Offset: 8224H
Power-On Default: Undefined

This register is used when the CPU is doing DMA transfers from video memory as specified by clearing bit 1 of MM8580 to 0 (read) and setting bit 0 of MM8588 to 1 (video DMA enable).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	DMA READ STRIDE									0	0	0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	DMA READ WIDTH									0	0	0

Bits 1-0 ATC - Address Tiling Control
00 = CPU address lines 14-8 are sequential (linear addressing)
01 = CPU address lines 14-8 are rearranged to 14, 9, 8, 13, 12, 11, 10
10 = CPU address lines 14-8 are rearranged to 10, 9, 8, 14, 13, 12, 11
11 = CPU address lines 14-8 are sequential (linear addressing)

The rearranged settings provide a narrower and deeper memory page size to minimize page breaks when drawing is limited to a small area of the screen. The 10 setting provides the narrowest page size.

Bit 2 Reserved = 0

Bits 11-3 DMA READ STRIDE

Value = Number of quadwords to add to the address at the end of a line to generate the address for the next line to be transferred

A DMA transfer from video memory to system memory starts at the address specified in MM8220_22_3 and proceeds for the number of quadwords defined by the value in bits 27-19 of this register. The stride value is then added to end of line address to get the address for the start of the next line to be transferred.

Bits 15-12 Reserved

Bits 18-16 Reserved = 0

Bits 27-19 DMA READ WIDTH

Value = [Number of quadwords per line to transfer to system memory] - 1

Bits 31-28 Reserved

Section 22: Miscellaneous Register Descriptions

In all register bit descriptions, the letter "R" identifies reserved bits (a reserved bit's read value is undefined unless noted, and you may write only zero to a reserved bit).

Subsystem Status Register (MM8504)

Read Only Offset: 8504H
Power-On Default: Undefined

This read-only register provides information on interrupt status, monitor I.D. and the number of bits per pixel. See the Subsystem Control (MM8504, Write Only) register for details on enabling and clearing interrupts.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	S3d ENG	S3D FIFO SLOTS FREE					LPB INT	3DF FIFO	CD DON	HD DON	FIFO EMP	FIFO OVF	3D DON	VSY INT
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 VSY INT - Vertical Sync Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bit 1 3D DON - S3d Engine Done Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bit 2 FIFO OVF - Command FIFO Overflow Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bit 3 FIFO EMP - Command FIFO Empty Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bit 4 HD DON - Host DMA Done Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bit 5 CD DON - Command DMA Done Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bit 6 3DF FIF - S3d FIFO Empty Status
0 = No interrupt
1 = Interrupt generated

Bit 7 LPB INT - LPB Interrupt Status
0 = No interrupt
1 = Interrupt generated

Bits 12-8 S3d FIFO SLOTS FREE
00000 = 0 slots free
.
.
.
10000 = 16 slots free (S3d FIFO is 16 slots deep)

Bit 13 S3d ENG - S3d Engine Status
0 = S3d Engine busy
1 = S3d Engine idle

Bits 31–14 Reserved

Subsystem Control Register (MM8504)

Write Only Offset: 8504H
Power-On Default: Undefined

This register allows each of several interrupt sources to be enabled or disabled. Interrupt status (Subsystem Status (MM8504, Read Only) can be cleared. This register also controls the software reset of the graphics engine.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S3d RST 1	0	3DF ENB	CDD ENB	FIFO EMP	ENB OVF	3DD ENB	VSY ENB	HDD ENB	3DF CLR	CDD CLR	HDD CLR	FIFO CLE	FIFO CLO	3DD CLR	VSY CLR
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 VSY CLR - Clear Vertical Sync Interrupt Status
0 = No change
1 = Clear

Bit 1 3DD CLR - Clear S3d Engine Done Interrupt Status
0 = No change
1 = Clear

Bit 2 FIFO CLO - Clear Command FIFO Overflow Interrupt Status
0 = No change
1 = Clear

Bit 3 FIFO CLE - Clear Command FIFO Empty Interrupt Status
0 = No change
1 = Clear

Bit 4 HDD CLR - Clear Host DMA Done Interrupt Status
0 = No change
1 = Clear

Bit 5 CDD CLR - Clear Command DMA Done Interrupt Status
0 = No change
1 = Clear

Bit 6 3DF CLR - Clear S3d FIFO Empty Interrupt Status
0 = No change
1 = Clear

Bit 7 HDD ENB - Host DMA Done Interrupt Enable
0 = Disable
1 = Enable interrupt when a host DMA transfer is complete and CR32_4 = 1

Bit 8 VSY ENB - Vertical Sync Interrupt Enable
0 = Disable
1 = Enable interrupt when VSYNC goes active and CR32_4 = 1

Bit 9 3DD ENB- S3d Engine Done Interrupt Enable
0 = Disable
1 = Enable interrupt when the S3D Engine completes its current task and becomes idle and CR32_4 = 1

Bit 10 FIFO ENB OVF - Command FIFO Overflow Interrupt Enable
0 = Disable
1 = Enable interrupt when the command FIFO overflows and CR32_4 = 1

Bit 11 FIFO ENB EMP - Command FIFO Empty Interrupt Enable
0 = Disable
1 = Enable interrupt when the command FIFO becomes empty and CR32_4 = 1

Bit 12 CDD ENB - Command DMA Done Interrupt Enable
0 = Disable
1 = Enable interrupt when a command DMA transfer is complete and CR32_4 = 1

Bit 13 3DF ENB - S3d FIFO Empty Interrupt Enable
0 = Disable
1 = Enable interrupt when the S3d FIFO becomes empty and CR32_4 = 1

Bits 15–14 S3d RST - S3d Engine Software Reset
00 = No change
01 = S3d Engine enabled
10 = Reset
11 = Reserved

Setting CR66_1 to 1 is equivalent to setting these bits to 10b.

Bits 31-16 Reserved

Advanced Function Control Register (MM850C)

Read/Write Offset: 850CH
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	LA ENB	R	R	RST DM	ENB EHFC
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 ENB EHFC - Enable Enhanced Functions
0 = Enable VGA and VESA planar (4 bits/pixel) modes
1 = Enable all other modes (Enhanced and VESA non-planar)

This bit is ORed with bit 0 of CR66 and is equivalent to it.

Bit 1 RST DM - Reset Read DMA (Read/Write)
0 = No effect
1 = Reset read DMA pointers

This bit should be toggled (program a 1 and then a 0) by software immediately after the completion of each read DMA operation.

Bits 3-2 Reserved

Bit 4 LA ENB- Linear Addressing Enable
0 = Disable linear addressing
1 = Enable linear addressing

This bit is ORed with bit 4 of CR58 and is equivalent to it.

Bits 31–5 Reserved

Section 23: DMA Register Descriptions

This section describes the Direct Memory Access (DMA) registers for ViRGE/VX. These registers are used to control the two DMA channels when ViRGE/VX operates as a PCI bus master. The video/graphics data transfer channel handles:

- Compressed video data transfers from system memory to an MPEG-1 decoder via the LPB
- Decompressed video data (software MPEG) transfers to the frame buffer via the LPB
- Frame buffer data transfers to system memory

For the latter case, the video memory read data location and structure are specified in MM8220 and MM8224. These are described in the Memory Port Controller section.

The command data channel handles transfers of command and drawing parameter data from system memory to the S3D Engine.

These two channels can operate independently.

In all register bit descriptions, the letter "R" identifies reserved bits (a reserved bit's read value is undefined unless noted, and you may write only zero to a reserved bit).

23.1 VIDEO/GRAPHICS DATA TRANSFER CHANNEL

Video DMA Starting System Memory Address Register (MM8580)

Read/Write Offset: 8580H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STARTING MEMORY ADDRESS														R/W	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
STARTING MEMORY ADDRESS															

Bit 0 Reserved

Bit 1 R/W - Video/Graphics DMA Read/Write
0 = Video DMA write (system memory to the LPB output FIFO)
1 = Video DMA read (video memory to system memory)

Data written to the LPB output FIFO can be directed to an MPEG decoder (compressed data) or to video memory with optional decimation.

Bits 31-2 STARTING MEMORY ADDRESS

Value = Starting memory address when performing a DMA transfer from video memory to system memory or from system memory to the LPB output FIFO

Video DMA Transfer Length Register (MM8584)

Read/Write Offset: 8584H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMA TRANSFER LENGTH														R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	DMA TRANSFER LENGTH							

Bits 1-0 Reserved

Bits 23-2 DMA TRANSFER LENGTH

Value = (Number of double words to transfer) - 1.

Bits 31-24 Reserved

Video DMA Transfer Enable Register (MM8588)

Read/Write Offset: 8588H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	VDE
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 VDE - Video/Graphics DMA Enable
0 = Disable video/graphics DMA
1 = Enable video graphics DMA

This bit is reset to 0 by the DMA controller at the completion of a video/graphics DMA transfer.

Bits 31-1 Reserved

23.2 COMMAND TRANSFER CHANNEL

Command DMA Base Address Register (MM8590)

Read/Write Offset: 8590H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASE ADDRESS				R	R	R	R	R	R	R	R	R	R	BS	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BASE ADDRESS															

Bit 0 Reserved

Bit 1 BS - Command DMA Buffer Size
0 = 4 KByte buffer size
1 = 64 KByte buffer size

Bits 11-2 Reserved

Bits 31-12 BASE ADDRESS

Value = Command DMA buffer base address

Bits 15-12 must be 000b for a 64K buffer size (64K aligned).

Command DMA Write Pointer Register (MM8594)

Read/Write Offset: 8594H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRITE POINTER														R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	WPU

Bits 1-0 Reserved

Bits 15-2 WRITE POINTER

Value = next doubleword address after the last doubleword written to the system memory buffer

Bit 16 WPU - Write Pointer Updated

Software must set this bit to 1 each time it updates the write pointer.

Bits 31-17 Reserved

Command DMA Read Pointer Register (MM8598)

Read/Write Offset: 8598H
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ POINTER														R	R
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 1-0 Reserved

Bits 15-2 READ POINTER

Value = Address of next doubleword in system memory to be read by the DMA

4K buffer: address = base address 31-12 (concat) read pointer 11-2 (concat) 00
64K buffer: address = base addess 31-16 (concat) read pointer 15-2 (concat) 00

After this pointer value is initialized, it is is updated automatically by ViRGE.

Bits 31-16 Reserved

Command DMA Enable Register (MM859C)

Read/Write Offset: 859CH
Power-On Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	CDE
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 CDE - Command DMA Enable
0 = Command DMA Disabled
1 = Command DMA Enabled

Bits 31-1 Reserved

Section 24: Local Peripheral Bus Register Descriptions

This section describes the Local Peripheral Bus (LPB) registers.

In all register bit descriptions, the letter "R" identifies reserved bits (a reserved bit's read value is undefined unless noted, and you may write only zero to a reserved bit).

LPB Mode Register (MMFF00)

Read/Write Address: FF00H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	LBA	CHS	CVS	LHS	LVS	R	R	CBS	SF	LR	LPB MODE			LE
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
CFL	R	R	R	R	ILC	SNO	CS	R	VFT		R	R	R	MBS	

Bit 0 LE - LPB Enable
0 = LPB Disabled
1 = LPB Enabled

Once enabled, the LPB is reset either by a system reset or via bit 4 of this register.

Bits 3-1 LPB MODE
000 = Scenic/MX2 Mode. The $\overline{\text{VREQ}}$/VRDY, $\overline{\text{CREQ}}$/CRDY, LCLK and LD[7:0] functions become active.
001 = Video 16 Mode (PCI only). The HS, VS, LCLK and LD[15:0] functions become active. ViRGE/VX expects 16-bit Philips digitizer input.
010 = Video 8 In Mode. The HS and VS function become active and ViRGE/VX expects video data in 8-bit units (LD[7:0]).
100 = Pass-through Mode. 32-bit data from the output FIFO is passed directly to the decimation input to the video FIFO. This allows decimation of CPU-provided data.

All other values are reserved.

Bit 4 LR- LPB Reset
0 = No effect
1 = Reset LPB

This bit should be set and then reset before switching between LPB modes.

Bit 5 SF - Skip Frames
0 = Write all received frames to memory
1 = Write every other received frame to memory (1, 3, etc.)

Bit 6 CBS - Color Byte Swap
0 = Incoming video is in U_{01}, Y_0, V_{01}, Y_1 format, byte swapping enabled
1 = Incoming video is in Y_0, U_{01}, Y_1, V_{01} format (e.g., SAA7110), no byte swapping

Bits 8-7 Reserved

Bit 9 LVS - LPB Vertical Sync Input Polarity
0 = LPB vertical sync input is active low
1 = LPB vertical sync input is active high

Bit 10 LHS - LPB Horizontal Sync Input Polarity
0 = LPB horizontal sync input is active low
1 = LPB horizontal sync input is active high

Bit 11 CVS - CPU VSYNC (Write Only)

Writing a 1 to this bit causes ViRGE/VX to do whatever functions it is programmed to do upon receipt of a VSYNC. For example, values programmed in certain registers only take effect at the next VSYNC.

Bit 12 CHS - CPU HSYNC (Write Only)

Writing a 1 to this bit causes ViRGE/VX to do whatever functions it is programmed to do upon receipt of an HSYNC.

Bit 13 LBA - Load Base Address (Write Only)

Writing a 1 to this bit immediately loads the base address currently being pointed to.

Bits 15-14 Reserved

Bits 17-16 MBS - Maximum LPB to Scenic/MX2 Compressed Data Burst Size (Scenic/MX2 mode only)
00 = Burst 1 32-bit word
01 = Burst 2 32-bit words
10 = Burst 3 32-bit words
11 = Burst all 32-bit words (until empty)

With a setting of 11b, software must ensure that no more than eight 32-bit words are burst to the Scenic/MX2 in a single burst. For example, if the FIFO is full (8 entries), no more entries should be written until the burst is complete.

Bits 20-18 Reserved

Bits 22-21 VFT - Video FIFO Threshold
00 = 1 FIFO slot
01 = 3 FIFO slots
10 = 5 FIFO slots
11 = 7 FIFO slots

When this many slots are filled in the video FIFO, a request is generated to the memory manager to begin emptying the FIFO. This is used to maximize the efficiency of the memory interface.

Bit 23 Reserved

Bit 24 CS - LPB Clock Source
0 = LPB clock driven by SCLK
1 = LPB clock driven by LCLK

Bit 25 SNO - Sync Non-Overlap
0 = No effect
1 = Don't add stride after first HSYNC

This bit must be set when the first HSYNC does not occur within the VSYNC active period.

Bit 26 ILC - Invert LCLK
0 = Use LCLK as received
1 = Invert the LCLK input

Bit 24 of this register must be set to 1 for this bit to be effective.

Bits 31-27 Reserved

LPB FIFO Status Register (MMFF04)

Read Only Address: FF04H
Power-on Default: 00000008H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	OFAE	OFE	OFF	R	R	R	R	R	R	R	OFIFO STATUS			
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
VF1AE	VF1E	VF1F	R	R	R	R	R	R	VF0AE	VF0E	VF0F	R	R	R	R

Bits 3-0 LPB Output FIFO Status
0000 = 0 FIFO slots free
0001 = 1 FIFO slot free
0010 = 2 FIFO slots free
0011 = 3 FIFO slots free
0100 = 4 FIFO slots free
0101 = 5 FIFO slots free
0110 = 6 FIFO slots free
0111 = 7 FIFO slots free
1000 = 8 FIFO slots free

Each slot contains 4 bytes

Bits 10-4 Reserved

Bit 11 OFF - LPB Output FIFO Full
0 = Output FIFO not full
1 = Output FIFO full

Bit 12 OFE - LPB Output FIFO Empty
0 = Output FIFO not empty
1 = Output FIFO empty

Bit 13 OFAE - LPB Output FIFO Almost Empty
0 = Output FIFO has something other than 1 slot filled
1 = Output FIFO has one slot filled

Bits 19-14 Reserved

Bit 20 VF0F - LPB Video FIFO 0 Full
0 = Video FIFO 0 not full
1 = Video FIFO 0 full

Bit 21 VF0E - LPB Video FIFO 0 Empty
0 = Video FIFO 0 not empty
1 = Video FIFO 0 empty

Bit 22 VF0AE - LPB Video FIFO 0 Almost Empty
0 = Video FIFO 0 has something other than 1 slot filled
1 = Video FIFO 0 has one slot filled

Bits 28-23 Reserved

Bit 29 VF1F - LPB Video FIFO 1 Full
0 = Video FIFO 1 not full
1 = Video FIFO 1 full

Bit 30 VF1E - LPB Video FIFO 1 Empty
0 = Video FIFO 1 not empty
1 = Video FIFO 1 empty

Bit 31 VF1AE - LPB Video FIFO 1 Almost Empty
0 = Video FIFO 1 has something other than 1 slot filled
1 = Video FIFO 1 has one slot filled

LPB Interrupt Flags Register (MMFF08)

Read/Write Address: FF08H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	SPS	EFI	ELI	FEI
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	SPW	R	R	R	R	SPM	EFM	ELM	FEM

Bit 0 FEI - LPB Output FIFO Empty Interrupt Status
0 = No interrupt
1 = LPB output FIFO empty

Writing a 1 to this bit clears the interrupt.

Bit 1 ELI - End of Line Interrupt Status
0 = No interrupt
1 = HSYNC input on pin B5

Writing a 1 to this bit clears the interrupt.

Bit 2 EFI - End of Frame Interrupt Status
0 = No interrupt
1 = VSYNC input on pin A5

Writing a 1 to this bit clears the interrupt.

Bit 3 SPS - Serial Port Start Detect Interrupt Status
0 - No interrupt
1 = Serial port start condition detected

A serial port start condition occurs when SPD is driven low by another device while SPCLK is not being driven low. Writing a 1 to this bit clears the interrupt.

Bits 15-4 Reserved

Bit 16 FEM - LPB Output FIFO Empty Interrupt Enable Mask
0 = LPB output FIFO empty interrupt disabled
1 = LPB output FIFO empty interrupt enabled

Bit 17 ELM - End of Line Interrupt Enable Mask
0 = End of Line interrupt disabled
1 = End of Line interrupt enabled

Bit 18 EFM - End of Frame Interrupt Enable Mask
0 = End of frame interrupt disabled
1 = End of frame interrupt enabled

Bit 19 SPM - Serial Port Start Detect Interrupt Mask
0 = Serial port start detect interrupt disabled
1 = Serial port start detect interrupt enabled

Bits 23-20 Reserved

Bit 24 SPW - Serial Port Wait
0 = Release SPCLK to float high
1 = Drive SPCLK low upon receipt of a serial port start condition

Setting this bit to 1 enables serial port wait states until the CPU is ready to process the data.

Bit 31-25 Reserved

LPB Frame Buffer Address 0 Register (MMFF0C)

Read/Write Address: FF0CH
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPB BUFFER ADDRESS 0															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	LPB BUFFER ADDRESS 0					

Bits 21-0 LPB Frame Buffer Address 0

Value = starting address 0 (offset in bytes from the start of the frame buffer) for writing LPB data to the frame buffer

This value will normally be the same as the secondary stream frame buffer address 0. The value must start on an 8-byte boundary.

Bits 31-22 Reserved

LPB Frame Buffer Address 1 Register (MMFF10)

Read/Write Address: FF10H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPB BUFFER ADDRESS 1															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	LPB BUFFER ADDRESS 1					

Bits 21-0 LPB Frame Buffer Address 1

Value = starting address 1 (offset in bytes from the start of the frame buffer) for writing LPB data to the frame buffer

This value will normally be the same as the secondary stream frame buffer address 1. Both address 0 and address 1 are defined when double buffering is used. The value must start on an 8-byte boundary.

Bits 31-22 Reserved

LPB Direct Read/Write Address Register (MMFF14)

Read/Write Address: FF14H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPB DIRECT READ/WRITE ADDRESS															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	TT			LPB READ/WRITE ADDRESS				

Bits 20-0 LPB Direct Read/Write Address

Value = address of Scenic/MX2 register to read/write

Bits 23-21 TT - Transaction Type (Scenic/MX2)
000 = Register write
001 = Register read
110 = Compressed video data write from the output FIFO. This value is automatically generated by hardware when data is written to the output FIFO.

Bits 31-24 Reserved

LPB Direct Read/Write Data Register (MMFF18)

Read/Write Address: FF18H
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPB DIRECT READ/WRITE DATA															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
LPB DIRECT READ/WRITE DATA															

Bits 31-0 LPB Direct Read/Write Data

A write to this register triggers a read/write sequence based on the address information in MMFF14_23-0.

LPB General Purpose Input/Output Port Register (MMFF1C)

Read/Write - see bit definitions Address: FF1CH
Power-on Default: Undefined

This register is available only for PCI bus configurations.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	LPB GIP				LPB GOP			
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits 3-0 LPB General Purpose Output Data Port

These bits are driven onto the LPB LD[3:0] lines whenever a write is performed to CR5C. $\overline{\text{STWR}}$ is asserted (low) at this time for use as an enable strobe for latching the data into an external buffer.

Bits 7-4 LPB General Purpose Input Data Port (Read only)

Whenever a write is performed to CR5C, $\overline{\text{STWR}}$ is asserted (low). This strobe can be used to enable a register to drive data onto any or all of the LD[7:4] lines. This data is then latched into these bits.

Bits 31-8 Reserved

Serial Port Register (MMFF20)

See Bit Definitions Address: FF20H
Power-on Default: 00000000H

This register can also be accessed at I/O ports E2H or E8H. See the Serial Communications Port description in Section 12.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	B4M	B3M	B2M	B1M	B0M	R	R	R	SPE	SDR	SCR	SDW	SCW
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit 0 SCW - Serial Clock Write
0 = Pin C6 is driven low
1 = Pin C6 is tri-stated

Pin C6 carries the DDC/I^2C clock, depending on the operational mode. When pin C6 is tri-stated, other devices may drive this line. The actual state of the pin is read via bit 2 of this register.

Bit 1 SDW - Serial Data Write
0 = Pin B4 is driven low
1 = Pin B4 is tri-stated

Pin B4 carries the DDC/I^2C data, depending on the operational mode. When pin B4 is tri-stated, other devices may drive this line. The actual state of the pin is read via bit 3 of this register.

Bit 2 SCR - Serial Clock Read (Read Only)
0 = Pin C6 is low
1 = Pin C6 is tri-stated (no device is driving this line)

Bit 3 SDR - Serial Data Read (Read Only)
0 = Pin B4 is low
1 = Pin B4 is tri-stated (no device is driving this line)

Bit 4 SPE - Serial Port Enable
0 = Use of bits 1-0 of this register disabled
1 = Use of bits 1-0 of this register enabled

Bits 5-7 Reserved

Bit 8 B0M - Bit 0 Mirror (Read Only)
0 = Pin C6 is driven low
1 = Pin C6 is tri-stated

Bit 9 B1M - Bit 1 Mirror (Read Only)
0 = Pin B4 is driven low
1 = Pin B4 is tri-stated

Bit 10 B2M - Bit 2 Mirror (Read Only)
0 = Pin C6 is low
1 = Pin C6 is tri-stated (no device is driving this line)

Bit 11 B3M -Bit 3 Mirror (Read Only)
0 = Pin B4 is low
1 = Pin B4 is tri-stated (no device is driving this line)

Bit 12 B4M - Bit 4 Mirror (Read Only)
0 = Use of bits 1-0 of this register disabled
1 = Use of bits 1-0 of this register enabled

Bits 31-13 Reserved

LPB Video Input Window Size Register (MMFF24)

Read/Write Address: FF24H
Power-on Default: Undefined

This register applies only to Video 8 In or Video 16 mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	VIDEO INPUT LINE WIDTH											
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	VIDEO INPUT WINDOW HEIGHT								

Bits 11-0 Video Input Line Width

Value = [# pixels x 2] - 2 for Video 8 mode
Value = # pixels -2 for Video 16 mode

This is the width of the displayed line after the offset specified in MMFF28_11-0. Before the 2 is subtracted, the number of pixels must be a multiple of 8. For example, in Video 16 mode, if the line width is 637 pixels, this must be rounded up to 640. The programmed value is then 640 - 2 = 638.

Bits 15-12 Reserved

Bits 24-16 Video Input Window Height

Value = [height in lines of each video input frame] - 1

This is the number of displayed lines - 1 after the offset specified in MMFF28_24_16.

Bits 31-25 Reserved

LPB Video Data Offsets Register (MMFF28)

Read/Write Address: FF28H
Power-on Default: Undefined

This register applies only to Video 8 In or Video 16 mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	HORIZONTAL VIDEO DATA OFFSET											
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
R	R	R	R	R	R	R	VERTICAL VIDEO DATA OFFSET								

Bits 11-0 Horizontal Video Data Offset

Value = [number of LCLKs between HSYNC and the start of valid pixel data] - 2

Bits 15-12 Reserved

Bits 24-16 Vertical Video Data Offset

Value = number of HSYNCs between VSYNC and the first valid data line

Bits 31-25 Reserved

LPB Horizontal Decimation Control Register (MMFF2C)

Read/Write Address: FF2CH
Power-on Default: Undefined

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VIDEO DATA LUMA MASK															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
VIDEO DATA LUMA MASK															

Bits 31-0 Video Data Luma Mask

Horizontal decimation is used only for YUV 4:2:2 data. Each 64 bytes of video data input is compared with this mask, with each bit corresponding to a YU or YV pair. If a bit in this mask is 1, the corresponding Y (luma) is discarded. If a bit is a 0, the corresponding luma is passed to the video memory. Kept lumas are paired sequentially and are assigned UVs (chromas) from the first luma of the pair. Normally, decimation starts with bit 0 after an HSYNC. If a horizontal video data offset is specified in MMFF28_11-0 (video 8 or 16 modes only), decimation aligns with the start of data after the offset.

LPB Vertical Decimation Control Register (MMFF30)

Read/Write Address: FF30H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VIDEO DATA LINE MASK															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VIDEO DATA LINE MASK															

Bits 31-0 Video Data Line Mask

Each 32 lines of video data input is compared with this mask. If a bit in the mask is 0, the corresponding line is discarded. If a bit is a 1, the corresponding line is passed to the video memory. If a vertical video data offset is specified in MMFF28_24-16 (video 8 or 16 modes only), decimation does not align with the starting line after the offset and instead starts from VSYNC.

LPB Line Stride (MMFF34)

Read/Write Address: FF34H
Power-on Default: 00000000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	LINE STRIDE											
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	R	R	R	R		R	R	R	R	R	R	R	R	R	R

Bits 11-0 Line Stride

Value = byte offset of vertically adjacent pixels

This offset is added to the line starting address each HSYNC to get the new line starting address. Each line must begin on an 8-byte boundary.

Bits 31-12 Reserved

LPB Output FIFO Register (MMFF40)

Read/Write Address: FF40H, FF44H...,FF5CH
Power-on Default: 00000000H

Writes to any of the addresses in this 8 doubleword address range will be transferred to the LPB input FIFO. This allows efficient use of the MOVSD assembly language instruction. Accesses must be to doubleword addresses.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OUTPUT FIFO DATA															
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
OUTPUT FIFO DATA															

Bits 31-0 Output FIFO Data

Note: Software must never transfer more compressed data than there is room for in the output FIFO. This information is read from MMFF04_3-0.

Section 25: PCI Register Descriptions

The PCI specification defines a configuration register space. These registers allow device relocation, device independent system address map construction and automatic configurations. ViRGE/VX provides a subset of these registers, which are described below.

The configuration register space occupies 256 bytes. When a configuration read or write command is issued, the AD[7:0] lines contain the address of the register in this space to be accessed. ViRGE/VX supports or returns 0 for the first 64 bytes of this space.

In the following register descriptions, 'R' stands for reserved (write = 0, read = undefined).

Vendor ID

Read Only Address: 00H
Power-On Default: 5333H

This read-only register identifies the device manufacturer.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Vendor ID															

Bits 15–0 Vendor ID
This is hardwired to 5333H to identify S3 Incorporated.

Device ID

Read Only Address: 02H
Power-On Default: 883DH

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Device ID															

Bits 15–0 Device ID

Hardwired to 883DH.

Command

Read/Write Address: 04H
Power-On Default: 0000H (PCI); 0003H (VL)

This register controls which types of PCI cycles ViRGE/VX can generate and respond to.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	DAC SNP	r	R	BME	MA	I/O

Bit 0 I/O - Enable Response to I/O Accesses
0 = Response to I/O space accesses is disabled
1 = Response to I/O space accesses enabled

Bit 1 MA - Enable Response to Memory Accesses
0 = Response to memory space accesses is disabled
1 = Response to memory space accesses enabled

Bit 2 BME - Bus Master Operation Enable
0 = Bus master operation disabled
1 = Bus master operation enabled

Bits 4-3 Reserved

Bit 5 DAC SNP - RAMDAC Register Access Snooping
0 = ViRGE/VX claims and responds to all RAMDAC register access cycles
1 = ViRGE/VX performs RAMDAC register writes but does not claim the PCI cycle. RAMDAC register read accesses are performed by ViRGE/VX.

Bits 15–6 Reserved

Status

Read/Write Address: 06H
Power-On Default: 0200H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	RMA	RTA	R	DEVSEL		R	R	R	R	R	R	R	R	R

Bits 8–0 Reserved

Bits 10–9 DEVSEL - Device Select Timing
01 = Medium DEVSEL timing. (hardwired)

Bit 11 Reserved

Bit 12 RTA - Received Target Abort
0 = No effect
1 = Bus master transaction terminated with target-abort

Bit 13 RMA - Received Master Abort
0 = No effect
1 = Bus master transaction terminated with master-abort

Bits 15–14 Reserved

Class Code

Read Only Address: 08H
Power-On Default: 30000xxH

This register is hardwired to 30000xxH to specify that ViRGE/VX is a VGA-compatible display controller. The xx will vary by chip revision.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PROGRAMMING INTERFACE								REVISION ID							
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BASE CLASS CODE								SUB-CLASS							

Latency Timer

Read/Write Address: 0DH
Power-On Default: 00H

7	6	5	4	3	2	1	0
BM LATENCY TIMER					0	0	0

Bits 2-0 Reserved = 0

These are the 3 lsb's of the latency timer value, providing 8 clocks granularity.

Bits 7-3 BM LATENCY TIMER - Bus Master Latency Timer

Value = number of PCI clocks ViRGE/VX can keep its bus master grant without having it removed

These are the 5 msb's of this value. The three lsb's are 000b. This value is normally programmed by the system BIOS based in part on the requested value in bits 15-8 of 3EH.

Base Address 0

Read/Write Address: 12H (high) 10H (low)
Power-On Default: 7000 0000H (PCI), 0000 0000H (VL)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	PREF = 0	TYPE =00		MSI = 0
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BASE ADDRESS 0						R	R	R	R	R	R	R	R	R	R

Bit 0 MSI - Memory Space Indicator
0 = Base registers map into memory space (hardwired)

Bits 2–1 TYPE - Type of Address Relocation
00 = Locate anywhere in 32-bit address space (hardwired)

Bit 3 PREF - Prefetchable
0 = Does not meet the prefetchable requirements (hardwired)

Bits 25–4 Reserved

Bits 31–26 BASE ADDRESS 0

Value = upper 6 bits of the base address for accessing ViRGE/VX registers and memory via memory-mapped I/O

This field provides for address relocation. These bits map to system address bits 31-26. All other address bits (25-4) return 0 on read to specify that the ViRGE/VX requires a 64 MByte address space. This field is normally programmed by the system BIOS at reset and should not be changed by graphics software. Note that writes to CR59_7-2 will also update this field, so is the linear addressing base address is being changed, the programmer must do a read-modify-write to ensure that this field is not changed.

BIOS ROM Base Address

Read/Write Address: 32H (high) 30H (low)
Power-On Default: 000C 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	ADE
31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BIOS ROM BASE ADDRESS															

Bit 0 ADE - Address Decode Enable
0 = Accesses to the BIOS ROM address space defined in this register are disabled
1 = Accesses to the BIOS ROM address space defined in this register are enabled

Bits 15–1 Reserved

Bits 31–16 BIOS ROM BASE ADDRESS
These are the upper 16 bits of the BIOS ROM address.

Interrupt Line

Read/Write Address: 3CH
Power-On Default: 00H

This register contains interrupt line routing information written by the POST program during power-on initialization.

7	6	5	4	3	2	1	0
INTERRUPT LINE							

Bits 7–0 INTERRUPT LINE

Interrupt Pin

Read Only Address: 3DH
Power-On Default: 01H

This register is hardwired to a value of 1 to specify that $\overline{\text{INTA}}$ is the interrupt pin used.

7	6	5	4	3	2	1	0
INTERRUPT PIN							

Bits 7–0 INTERRUPT PIN

Latency/Grant

Read Only Address: 3EH
Power-On Default: FF04H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAXIMUM LATENCY								MINIMUM GRANT							

Bits 7–0 MINIMUM GRANT

Value = Length of burst period required in units of 250 ns (33 MHz clock)

Bits 15–8 MAXIMUM LATENCY

Value = Maximum latency of PCI access in units of 250 ns (33 MHz clock)

Appendix A: Listing of Raster Operations

ViRGE/VX supports all 256 triadic raster operations (ROPs) for BitBLTs as defined by Microsoft for Windows. The coding for these is found on the following pages.

The HEX value in the first column is the ROP code. This value must be programmed into bits 7-0 of D2E8H at the time that a ROPBLT command is executed.

The effect of the ROP is shown in reverse Polish notation in the second column. This is interpreted as follows:

S = Source bitmap

P = Pattern

D = Destination bitmap

The source bitmap can be either the CPU or the current screen, as specified by bit 7 of the Command Set register. A CPU source can be either monochrome or color, as specified by bit 6 of the Command Set register. A screen source is always color.

The pattern may be either monochrome or color, as specified by bit 8 of the Command Set register.

The destination bitmap is always the screen. It is always color (as opposed to monochrome).

The boolean operators used as as follows:

o = bitwise OR

x = bitwise EXCLUSIVE OR

a = bitwise AND

n = bitwise NOT (inverse)

For example, ROP 16H is PSDPSanaxx. The pattern is first ANDed with the source [PSD(PaS)naxx]. The result is inverted and then ANDed with the destination [PS((Da(notPaS))xx. This result is EXCLUSIVE ORed with the source. Finally, the result of this is EXCLUSIVE ORed with the pattern.

Programming using ROPBLTs is explained in Enhanced Programming section.

HEX	In Reverse Polish
00	0
01	DPSoon
02	DPSona
03	PSon
04	SDPona
05	DPon
06	PDSxnon
07	PDSaon
08	SDPnaa
09	PDSxon
0A	DPna
0B	PSDnaon
0C	SPna
0D	PDSnaon
0E	PDSonon
0F	Pn
10	PDSona
11	DSon
12	SDPxnon
13	SDPaon
14	DPSxnon
15	DPSaon
16	PSDPSanaxx
17	SSPxDSxaxn
18	SPxPDxa
19	SDPSanaxn
1A	PDSPaox
1B	SDPSxaxn
1C	PSDPaox
1D	DSPDxaxn
1E	PDSox
1F	PDSoan
20	DPSnaa
21	SDPxon
22	DSna
23	SPDnaon
24	SPxDSxa
25	PDSPanaxn
26	SDPSaox
27	SDPSxnox
28	DPSxa
29	PSDPSaoxxn
2A	DPSana
2B	SSPxPDxaxn

HEX	In Reverse Polish
2C	SPDSoax
2D	PSDnox
2E	PSDPxox
2F	PSDnoan
30	PSna
31	SDPnaon
32	SDPSoox
33	Sn
34	SPDSaox
35	SPDSxnox
36	SDPox
37	SDPoan
38	PSDPoax
39	SPDnox
3A	SPDSxox
3B	SPDnoan
3C	PSx
3D	SPDSonox
3E	SPDSnaox
3F	PSan
40	PSDnaa
41	DPSxon
42	SDxPDxa
43	SPDSanaxn
44	SDna
45	DPSnaon
46	DSPDaox
47	PSDPxaxn
48	SDPxa
49	PDSPDaoxxn
4A	DPSDoax
4B	PDSnox
4C	SDPana
4D	SSPxDSxoxn
4E	PDSPxox
4F	PDSnoan
50	PDna
51	DSPnaon
52	DPSDaox
53	SPDSxaxn
54	DPSonon
55	Dn
56	DPSox
57	DPSoan

HEX	In Reverse Polish
58	PDSPoax
59	DPSnox
5A	DPx
5B	DPSDonox
5C	DPSDxox
5D	DPSnoan
5E	DPSDnaox
5F	DPan
60	PDSxa
61	DSPDSaoxxn
62	DSPDoax
63	SDPnox
64	SDPSoax
65	DSPnox
66	DSx
67	SDPSonox
68	DSPDSonoxxn
69	PDSxxn
6A	DPSax
6B	PSDPSoaxxn
6C	SDPax
6D	PDSPDoaxxn
6E	SDPSnoax
6F	PDSxnan
70	PDSana
71	SSDxPDxaxn
72	SDPSxox
73	SDPnoan
74	DSPDxox
75	DSPnoan
76	SDPSnaox
77	DSan
78	PDSax
79	DSPDSoaxxn
7A	DPSDnoax
7B	SDPxnan
7C	SPDSnoax
7D	DPSxnan
7E	SPxDSxo
7F	DPSaan
80	DPSaa
81	SPxDSxon
82	DPSxna
83	SPDSnoaxn

HEX	In Reverse Polish
84	SDPxna
85	PDSPnoaxn
86	DSPDSoaxx
87	PDSaxn
88	DSa
89	SDPSnaoxn
8A	DSPnoa
8B	DSPDxoxn
8C	SDPnoa
8D	SDPSxoxn
8E	SSDxPDxax
8F	PDSanan
90	PDSxna
91	SDPSnoaxn
92	DPSDPoaxx
93	SPDaxn
94	PSDPSoaxx
95	DPSaxn
96	DPSxx
97	PSDPSonoxx
98	SDPSonoxn
99	DSxn
9A	DPSnax
9B	SDPSoaxn
9C	SPDnax
9D	DSPDoaxn
9E	DSPDSaoxx
9F	PDSxan
A0	DPa
A1	PDSPnaoxn
A2	DPSnoa
A3	DPSDxoxn
A4	PDSPonoxn
A5	PDxn
A6	DSPnax
A7	PDSPoaxn
A8	DPSoa
A9	DPSoxn
AA	D
AB	DPSono
AC	SPDSxax
AD	DPSDaoxn
AE	DSPnao
AF	DPno

HEX	In Reverse Polish
B0	PDSnoa
B1	PDSPxoxn
B2	SSPxDSxox
B3	SDPanan
B4	PSDnax
B5	DPSDoaxn
B6	DPSDPaoxx
B7	SDPxan
B8	PSDPxax
B9	DSPDaoxn
BA	DPSnao
BB	DSno
BC	SPDSanax
BD	SDxPDxan
BE	DPSxo
BF	DPSano
C0	PSa
C1	SPDSnaoxn
C2	SPDSonoxn
C3	PSxn
C4	SPDnoa
C5	SPDSxoxn
C6	SDPnax
C7	PSDPoaxn
C8	SDPoa
C9	SPDoxn
CA	DPSDxax
CB	SPDSaoxn
CC	S
CD	SDPono
CE	SDPnao
CF	SPno
D0	PSDnoa
D1	PSDPxoxn
D2	PDSnax
D3	SPDSoaxn
D4	SSPxPDxax
D5	DPSanan
D6	PSDPSaoxx
D7	DPSxan
D8	PDSPxax
D9	SDPSaoxn
DA	DPSDanax
DB	SPxDSxan

HEX	In Reverse Polish
DC	SPDnao
DD	SDno
DE	SDPxo
DF	SDPano
E0	PDSoa
E1	PDSoxn
E2	DSPDxax
E3	PSDPaoxn
E4	SDPSxax
E5	PDSPaoxn
E6	SDPSanax
E7	SPxPDxan
E8	SSPxDSxax
E9	DSPDSanaxxn
EA	DPSao
EB	DPSxno
EC	SDPao
ED	SDPxno
EE	DSo
EF	SDPnoo
F0	P
F1	PDSono
F2	PDSnao
F3	PSno
F4	PSDnao
F5	PDno
F6	PDSxo
F7	PDSano
F8	PDSao
F9	PDSxno
FA	DPo
FB	DPSnoo
FC	PSo
FD	PSDnoo
FE	DPSoo
FF	1

Appendix B: Register Reference

This Appendix contains tables listing all the registers in each of categories corresponding to Sections 16-25 of this data book.

- VGA
- Extended Sequencer
- Extended CRTC
- S3d
- Miscellaneous
- Streams Processor
- Memory Port
- DMA
- LPB
- PCI Configuration Space

Within each table, registers are listed in order of increasing addresses/indices. Name, address, register bit descriptions with read/write status and the page number of the detailed register description are provided for each register. All addresses and indices are hexadecimal values.

B.1 VGA REGISTERS

? = B for monochrome, D for color.

Table B-1. VGA Registers

Address	Index Bit(s)		Register Name Bit Description	Description Page
General or External Registers				
3C2			**Miscellaneous Output**	**16-1**
	0	W	Color emulation. Address based at 3Dx	
	1	W	Enable CPU access of video memory	
	3-2	W	Video DCLK select. Enable DCLK PLL loading	
	4	W	Reserved	
	5	W	Select the high 64K page of memory	
	6	W	Make HSYNC an active low signal	
	7	W	Make VSYNC an active low signal	
3CC			**Miscellaneous Output**	**16-1**
	0	R	Color emulation. Address based at 3Dx	
	1	R	Enable CPU access of video memory	
	3-2	R	Video DCLK select. Enable DCLK PLL loading	
	4	R	Reserved	
	5	R	Select the high 64K page of memory	
	6	R	Make HSYNC an active low signal	
	7	R	Make VSYNC an active low signal	
3?A			**Feature Control**	**16-2**
	2-0	W	Reserved	
	3	W	VSYNC is ORed with the internal display enable signal	
	7-4	W	Reserved	
3CA			**Feature Control**	**16-2**
	2-0	R	Reserved	
	3	R	VSYNC is ORed with the internal display enable signal	
	7-4	R	Reserved	
3C2			**Input Status 0**	**16-3**
	3-0	R	Reserved	
	4	R	The internal **SENSE** signal is a logical 1	
	6-5	R	Reserved	
	7	R	Vertical retrace interrupt to the CPU is pending	

Table B-1. VGA Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?A			**Input Status 1**	**16-3**
	0	R	The display in not in active display mode	
	1	R	Reserved	
	2	R	Reserved = 1	
	3	R	Vertical retrace period is active	
	5-4	R	Feedback of two color outputs for test purposes	
	7-6	R	Reserved	
3C3			**Video Subsystem Enable**	**16-4**
	0	R/W	Enable VGA display	
	7-1	R/W	Reserved	
Sequencer Registers				
3C4			**Sequencer Index**	**16-5**
	4-0	R/W	Index to the sequencer register to be accessed	
	7-5	R/W	Reserved	
3C5			**Sequencer Data**	**16-5**
	7-0	R/W	Data to or from the sequencer register accessed	
3C5	**00**		**Reset (SR0)**	**16-6**
	0	R/W	Asynchronous reset (not functional for ViRGE	
	1	R/W	Synchronous reset (not functional for ViRGE)	
	7-2	R/W	Reserved	
3C5	**01**		**Clocking Mode (SR1)**	**16-6**
	0	R/W	Character clocks are 8 dots wide	
	1	R/W	Reserved	
	2	R/W	Load the video serializers every second character clock	
	3	R/W	The internal character clock is 1/2 the DCLK frequency	
	4	R/W	Load the video serializers every fourth character clock	
	5	R/W	Screen is turned off	
3C5	**02**		**Enable Write Plane (SR2)**	**16-7**
	3-0	R/W	Enables a CPU write to the corresponding color plane	
	7-4	R/W	Reserved	
3C5	**03**		**Character Font Select (SR3)**	**16-8**
	4, 1-0	R/W	Select Font B	
	5,3-2	R/W	Select Font A	
	7-6	R/W	Reserved	
3C5	**04**		**Memory Mode Control (SR4)**	**16-9**
	0	R/W	Reserved	
	1	R/W	Memory access to 256K allowed (required for VGA)	
	2	R/W	Sequential addressing for CPU video memory accesses	

Table B-1. VGA Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
CRT Controller Registers				
3?4			**CRT Controller Index**	**16-10**
	7-0	R/W	Index to the CRTC register to be accessed	
3?5			**CRT Controller Data**	**16-10**
	7-0	R/W	Data to or from the CRTC register accessed	
3?5	**00**		**Horizontal Total (CR0)**	**16-11**
	7-0	R/W	Number of characters in a line -5	
3?5	**01**		**Horizontal Display End (CR1)**	**16-11**
	7-0	R/W	One less than the total number of displayed characters	
3?5	**02**		**Start Horizontal Blank (CR2)**	**16-12**
	7-0	R/W	Character count where horizontal blanking starts	
3?5	**03**		**End Horizontal Blank (CR3)**	**16-12**
	4-0	R/W	End position of horizontal blanking	
	6-5	R/W	Display enable skew in character clocks	
	7	R/W	Reserved	
3?5	**04**		**Start Horizontal Sync Position (CR4)**	**16-13**
	7-0	R/W	Character count where HSYNC goes active	
3?5	**05**		**End Horizontal Sync Position (CR5)**	**16-13**
	4-0	R/W	Position where HSYNC goes inactive	
	6-5	R/W	Horizontal retrace end delay in character clocks	
	7	R/W	End horizontal blanking bit 5	
3?5	**06**		**Vertical Total (CR6)**	**16-14**
	7-0	R/W	Number of lines - 2	
3?5	**07**		**CRTC Overflow (CR7)**	**16-14**
	0	R/W	Vertical total bit 8	
	1	R/W	Vertical display end bit 8	
	2	R/W	Vertical retrace start bit 8	
	3	R/W	Start vertical blank bit 8	
	4	R/W	Line compare bit 8	
	5	R/W	Vertical total bit 9	
	6	R/W	Vertical display end bit 9	
	7	R/W	Vertical retrace start bit 9	
3?5	**08**		**Preset Row Scan (CR8)**	**16-15**
	4-0	R/W	Line where first character row begins	
	6-5	R/W	Number of bytes to pan horizontally	
	7	R/W	Reserved	

Table B-1. VGA Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	09		**Maximum Scan Line (CR9)**	16-15
	4-0	R/W	Character height in scan lines -1	
	5	R/W	Start vertical blank bit 9	
	6	R/W	Line compare bit 9	
	7	R/W	Double scanning (repeat each line) enabled	
3?5	0A		**Cursor Start Scan Line (CRA)**	16-16
	4-0	R/W	Cursor starting line within the character cell	
	5	R/W	Turns off the cursor	
	7-6	R/W	Reserved	
3?5	0B		**Cursor End Scan Line (CRB**	16-16
	4-0	R/W	Cursor ending line within the character cell	
	6-5	R/W	Cursor skew to right in characters	
	7	R/W	Reserved	
3?5	OC		**Start Address High (CRC)**	16-17
	7-0	R/W	Bits 15-8 of the display start address	
3?5	OD		**Start Address Low (CRD)**	16-17
	7-0	R/W	Bits 7-0 of the display start address	
3?5	OE		**Cursor Location Address High (& Hardware Cursor Foreground Color in Enhanced Mode) (CRE)**	16-17
	7-0	R/W	Bits 15-8 of the cursor location start address	
3?5	OF		**Cursor Location Address Low (& Hardware Cursor Background Color in Enhanced Mode) (CRF)**	16-17
	7-0	R/W	Bits 7-0 of the cursor location start address	
3?5	10		**Vertical Retrace Start (CR10**	16-18
	7-0	R/W	Vertical retrace start in scan lines	
3?5	11		**Vertical Retrace End (CR11)**	16-18
	3-0	R/W	Vertical retrace end in scan lines	
	4	R/W	Clear the vertical retrace interrupt flip-flop	
	5	R/W	Disable vertical interrupts	
	6	R/W	Five RAM refresh cycles per horizontal line	
	7	R/W	Lock writes to CR0-CR7	
3?5	12		**Vertical Display End (CR12)**	16-19
	7-0	R/W	Number of scan lines of active video	
3?5	13		**Offset (CR13)**	16-20
	7-0	R/W	Memory start address jump from one scan line to the next	

Table B-1. VGA Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	**14**		**Underline Location (CR14)**	**16-20**
	4-0	R/W	Horizontal scan line where underline occurs	
	5	R/W	Memory address counter increment is 4 character clocks	
	6	R/W	Memory accessed as doublewords	
	7	R/W	Reserved	
3?5	**15**		**Start Vertical Blank (CR15)**	**16-21**
	7-0	R/W	Horizontal scan line where vertical blanking starts	
3?5	**16**		**End Vertical Blank (CR16**	**16-21**
	7-0	R/W	Horizontal scan line where vertical blanking ends	
3?5	**17**		**CRTC Mode Control (CR17)**	**16-22**
	0	R/W	Enable bank 2 mode for CGA emulation	
	1	R/W	Enable bank 4 mode for CGA emulation	
	2	R/W	Use horizontal retrace clock divided by 2	
	3	R/W	Enable count by 2 mode	
	4	R/W	Reserved	
	5	R/W	Enable CGA mode address wrap	
	6	R/W	Use byte address mode	
	7	R/W	Horizontal and vertical retrace signals enabled	
3?5	**18**		**Line Compare (CR18)**	**16-23**
	7-0	R/W	Line at which memory address counter cleared to 0	
3?5	**22**		**CPU Latch Data (CR22)**	**16-24**
	7-0	R	Value in the CPU latch in the graphics controller	
3?5	**24,26**		**Attribute Controller Flag/Index**	**16-24**
	5-0	R	Value of the attribute controller index data at 3C0H	
	6	R	Reserved	
	7	R	State of inverted internal address flip-flop	
Graphics Controller Registers				
3CE			**Graphics Controller Index**	**16-25**
	3-0	R/W	Index to the graphics controller register to be accessed	
	7-4	R/W	Reserved	
3CF			**Graphics Controller Data**	**16-25**
	7-0	R/W	Data to or from the graphics controller register accessed	
3CF	**00**		**Set/Reset (GR0)**	**16-26**
	3-0	R/W	Color value for CPU memory writes	
	7-4	R/W	Reserved	
3CF	**01**		**Enable Set/Reset (GR1)**	**16-26**
	3-0	R/W	Enable planes for writing GR0 data	
	7-4	R/W	Reserved	

Table B-1. VGA Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3CF	**02**		**Color Compare (GR2)**	**16-27**
	3-0	R/W	Reference color for color compare operations	
	7-4	R/W	Reserved	
3CF	**03**		**Raster Operation/Rotate Counter (GR3)**	**16-27**
	2-0	R/W	Number of right rotate positions for a CPU memory write	
	4-3	R/W	Select raster operation (logical function)	
	7-5	R/W	Reserved	
3CF	**04**		**Read Plane Select (GR4)**	**16-28**
	1-0	R/W	Select planes for reading	
	7-2	R/W	Reserved	
3CF	**05**		**Graphics Controller Mode (GR5)**	**16-29**
	1-0	R/W	Select write mode	
	2	R/W	Reserved	
	3	R/W	Enable read compare operation	
	4	R/W	Select odd/even addressing	
	5	R/W	Select odd/even shift mode	
	6	R/W	Select 256 color shift mode	
	7	R/W	Reserved	
3CF	**06**		**Memory Map Mode Control (GR6)**	**16-30**
	0	R/W	Select graphics mode memory addressing	
	1	R/W	Chain odd/even planes	
	3-2	R/W	Select memory mapping	
	7-4	R/W	Reserved	
3CF	**07**		**Color Don't Care (GR7)**	**16-31**
	3-0	R/W	Select color plane used for color comparison	
	7-4	R/W	Reserved	
3CF	**08**		**Bit Mask (GR8)**	**16-31**
	7-0	R/W	Each bit is a mask for the corresponding memory plane bit	
Attribute Registers				
3C0			**Attribute Controller Index**	**16-32**
	4-0	R/W	Index to the attribute controller register to be accessed	
	5	R/W	Enable video display	
	7-6	R/W	Reserved	
3C1/0			**Attribute Controller Data**	**16-33**
	7-0	R/W	Data to or from the attribute controller register accessed	
3C1/0	**00–0F**		**Palette Register (AR0–ARF)**	**16-33**
	5-0	R/W	Color value	
	7-6	R/W	Reserved	

Table B-1. VGA Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3C1/0	**10**		**Attribute Mode Control (AR10**	**16-34**
	0	R/W	Select graphics mode	
	1	R/W	Select monochrome display	
	2	R/W	Enable line graphics characters	
	3	R/W	Enable blinking	
	4	R/W	Reserved	
	5	R/W	Enable top panning	
	6	R/W	Select 256 color mode	
	7	R/W	Bits 5-4 of video output come from AR14_1-0	
3C1/0	**11**		**Border Color (AR11)**	**16-35**
	7-0	R/W	Border color value	
3C1/0	**12**		**Color Plane Enable (AR12)**	**16-35**
	3-0	R/W	Display plane enable	
	5-4	R/W	Select inputs to bits 5-4 of 3?AH	
	7-6	R/W	Reserved	
3C1/0	**13**		**Horizontal Pixel Panning (AR13)**	**16-36**
	3-0	R/W	Number of pixels to shift the display to the left	
	7-4	R/W	Reserved	
3C1/0	**14**		**Pixel Padding (AR14)**	**16-37**
	1-0	R/W	Bits 5-4 of the video output if AR10_7 = 1	
	3-2	R/W	Bits 7-6 of the video output	
	7-4	R/W	Reserved	
RAMDAC Registers				
3C6			**DAC Mask**	**16-38**
	7-0	R/W	Pixel read mask	
3C7			**DAC Read Index**	**16-38**
	7-0	W	Index to palette register to be read	
3C7			**DAC Status**	**16-39**
	1-0	R	Shows whether previous DAC cycle was a read or write	
	7-2	R	Reserved	
3C8	–		**DAC Write Index**	**16-40**
	7-0	R/W	Index to palette register to be written or General Input Port read data	
3C9			**DAC Data**	**16-41**
	7-0	R/W	Data from register pointed to by DAC Read or Write Index	

B.2 EXTENDED SEQUENCER REGISTERS

Table B-2. Extended Sequencer Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3C5	**08**		**Unlock Extended Sequencer (SR8)**	**17-1**
	7-0	R/W	Load xxxx0110b to unlock SR9-SR1C	
3C5	**09**	**R/W**	**Extended Sequencer 9 (SR9)**	**17-1**
	6-0	R/W	Reserved	
	7	R/W	Memory-mapped I/O only (no PIO)	
3C5	**0A**		**Extended Sequencer A (SRA)**	**17-2**
	4-0	R/W	Reserved	
	5	R/W	PD[63:0] not tri-stated	
	6	R/W	Reserved	
	7	R/W	2 MCLK memory writes	
3C5	**0B**		**Extended Sequencer B (SRB)**	**17-3**
	0	R/W	Use VCLKI for internal dot clock functions (test only)	
	1	R/W	Pixel data from VAFC latched by VCLKI	
	3-2	R/W	Reserved	
	7-4	R/W	Specify color mode for feature connector input	
3C5	**0D**		**Extended Sequencer D (SRD)**	**17-4**
	0	R/W	Enable feature connector operation	
	3-1	R/W	Reserved	
	5-4	R/W	HSYNC control for Green PC requirements	
	7-6	R/W	VSYNC control for Green PC requirements	
3C5	**10**		**MCLK Value Low (SR10)**	**17-5**
	4-0	R/W	MCLK N-divider value	
	6-5	R/W	MCLK R value	
	7	R/W	Reserved	
3C5	**11**		**MCLK Value High (SR11)**	**17-5**
	6-0	R/W	MCLK M-divider value	
	7	R/W	Reserved	
3C5	**12**		**DCLK Value Low (SR12)**	**17-6**
	4-0	R/W	DCLK N-divider value	
	6-5	R/W	DCLK R value	
	7	R/W	Reserved	
3C5	**13**		**DCLK Value High (SR13)**	**17-6**
	6-0	R/W	DCLK M-divider value	
	7	R/W	Reserved	

Table B-2. Extended Sequencer Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3C5	**14**		**CLKSYN Control 1 (SR14)**	**17-7**
	0	R/W	DCLK PLL powered down (test only)	
	1	R/W	MCLK PLL powered down (test only)	
	2	R/W	Enable internal clock synthesizer counter	
	3	R/W	Select MCLK or DCLK for testing	
	4	R/W	Clear clock synthesizer counter	
	5	R/W	Pin A16 tri-stated	
	6	R/W	MCLK is input on pin A16 (test only)	
	7	R/W	DCLK is input on pin C21 (test only)	
3C5	**15**		**CLKSYN Control 2 (SR15)**	**17-8**
	0	R/W	Load new MCLK frequency	
	1	R/W	Load new DCLK frequency	
	2	R/W	MCLK output on pin D17 (test only)	
	3	R/W	VCLK direction determined by EVCLK	
	4	R/W	Divide DCLK by 2	
	5	R/W	Load MCLK and DCLK PLL values immediately	
	6	R/W	Invert DCLK	
	7	R/W	Enable 2 MCLK memory writes	
3C5	**16**		**CLKSYN Test High (SR16)**	**17-9**
	7-0	R/W	High byte of clock synthesis test results	
3C5	**17**		**CLKSYN Test Low (SR17)**	**17-10**
	7-0	R/W	Low byte of clock synthesis test results	
3C5	**18**		**RAMDAC/CLKSYN Control (SR18)**	**17-10**
	0	R/W	RAMDAC test counter enabled (test only)	
	1	R/W	Reset RAMDAC test counter	
	2	R/W	Place red data on internal data bus (test only)	
	3	R/W	Place green data on internal data bus (test only)	
	4	R/W	Place blue data on internal data bus (test only)	
	5	R/W	Power-down RAMDAC	
	6	R/W	Select 1 cycle LUT write	
	7	R/W	Reserved	
3C5	**19**		**RAM Test (SR19)**	**17-11**
	0	R/W	RAM test enable (test only)	
	1	R	RAM test done	
	7-2	R/W	Reserved	

Table B-2. Extended Sequencer Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3C5	**1A**		**Extended Sequencer 1A (SR1A)**	**17-12**
	3-0	R/W	PLL VCO adjustment	
	4	R/W	Reserved	
	5	R/W	Add BLANK pedestal to RAMDAC output	
	7-6	R/W	Reserved	
3C5	**1C**		**Extended Sequencer 1C (SR1C)**	**17-13**
	1-0	R/W	Select functions for pins A16, A17 and B8	
	1	R/W	Reserved	
3C5	**22**		**DCLK0 Value Low (SR22)**	**17-13**
	5-0	R/W	DCLK0 N-divider value	
	7-6	R/W	DCLK0 R value	
3C5	**23**		**DCLK0 Value High (SR23)**	**17-14**
	7-0	R/W	DCLK0 M-divider value	
3C5	**24**		**DCLK1 Value Low (SR24)**	**17-14**
	5-0	R/W	DCLK1 N-divider value	
	7-6	R/W	DCLK1 R value	
3C5	**25**		**DCLK1 Value High (SR25)**	**17-15**
	7-0	R/W	DCLK1 M-divider value	

B.3 EXTENDED CRTC REGISTERS

Table B-3. Extended CRTC Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	2D		**Device ID High (CR2D)**	18-1
	7-0	R	High byte of device ID (88H)	
3?5	2E		**Device ID Low (CR2E)**	18-1
	7-0	R	Low byte of device ID (3DH)	
3?5	2F		**Revision (CR2F)**	18-2
	7-0	R	Revision level (subject to change)	
3?5	30		**Chip ID/Rev (CR30)**	18-2
	7-0	R	Old chip Identification - E1H	
3?5	31		**Memory Configuration (CR31)**	18-2
	0	R/W	Enable base address offset (CR6A_6-0)	
	1	R/W	Reserved	
	2	R/W	Enable VGA 16-Bit Memory Bus Width	
	3	R/W	Use Enhanced mode memory mapping	
	5-4	R/W	Old display start address bits 17-16 (see CR69_3-0)	
	6	R/W	Enable high speed text display font fetch mode	
	7	R/W	Reserved	
3?5	32		**Backward Compatibility 1 (CR32)**	18-3
	3-0	R/W	Reserved	
	4	R/W	Enable interrupt generation	
	5	R/W	Reserved	
	6	R/W	Use standard VGA memory wrapping at 256K boundary	
	7	R/W	Tri-state off SC, SE and DSF	
3?5	33		**Backward Compatibility 2 (CR33)**	18-4
	0	R/W	Reserved	
	1	R/W	Disable write protection provided by CR11_7 on CR7_1,6	
	2	R/W	Reserved	
	3	R/W	VCLK is internal DCLK	
	4	R/W	Disable writes to RAMDAC registers (3C6H-3C9H)	
	5	R/W	BLANK signal active during entire non-active video period	
	6	R/W	Disable writes to Palette/Overscan registers (AR0-ARF)	
	7	R/W	Reserved	

Table B-3. Extended CRTC Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	34		**Backward Compatibility 3 (CR34)**	18-5
	0	R/W	PCI DAC snoop method select	
	1	R/W	Disable PCI master abort handling during DAC snoop	
	2	R/W	Disable PCI retry handling during DAC snoop	
	3	R/W	Reserved	
	4	R/W	Enable Data Transfer Execute Position register (CR3B)	
	7-5	R/W	Reserved	
3?5	35		**CRT Register Lock (CR35)**	18-6
	3-0	R/W	Old CPU base address (see CR6A_6-0)	
	4	R/W	Lock Vertical Timing registers	
	5	R/W	Lock Horizontal Timing registers	
	7-6	R/W	Reserved	
3?5	36		**Configuration 1 (CR36)**	18-7
	1-0	R/W	Reserved	
	3-2	R/W	Select memory operation type	
	4	R/W	Reserved	
	6-5	R/W	Define display memory size	
	7	R/W	8-column block write support	
3?5	37		**Configuration 2 (CR37)**	18-7
	0	R/W	Reserved	
	1	R/W	Enable test mode	
	2	R/W	Reserved	
	3	R/W	Use internal MCLK, DCLK	
	4	R/W	Reserved	
	6-5	R/W	Specify DRAM size for mixed DRAM/VRAM configurations	
	7	R/W	Enable scan testing	
3?5	38		**Register Lock 1 (CR38)**	18-9
	7-0	R/W	Unlock S3 VGA registers (CR30-CR3C)	
3?5	39		**Register Lock 2 (CR39)**	18-9
	7-0	R/W	Unlock System Control, System Extension and Strapping registers (CR40-CR4F, CR50-CR6D)	
	7	R/W	Disable PCI bus read burst cycles	

Table B-3. Extended CRTC Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	**3A**		**Miscellaneous 1 (CR3A)**	**18-10**
	1-0	R/W	Select alternate refresh count per horizontal line	
	2	R/W	Enable alternate refresh count (CR3A_1-0)	
	3	R/W	Enable simultaneous VGA text and Enhanced modes	
	4	R/W	Enable 8-, 16- or 24/32-bit color Enhanced modes	
	5	R/W	Enable high speed text font writing	
	6	R/W	Reserved	
3?5	**3B**		**Data Transfer Execute Position (CR3B)**	**18-11**
	7-0	R/W	Specify VRAM data transfer start position	
3?5	**3C**		**Interlace Retrace Start (CR3C)**	**18-11**
	7-0	R/W	Specify interlaced mode retrace start position	
3?5	**40**		**System Configuration (CR40)**	**18-12**
	0	R/W	Enable Enhanced mode register access	
	4-1	R/W	Reserved	
	5	R/W	Reserved = 1	
	7-6	R/W	Reserved	
3?5	**41**		**BIOS Flag (CR41)**	**18-12**
	7-0	R/W	Used by the video BIOS	
3?5	**42**		**Mode Control (CR42)**	**18-13**
	4-0	R/W	Reserved	
	5	R/W	Select Interlaced mode	
	6	R/W	Reserved	
3?5	**43**		**Extended Mode (CR43)**	**18-13**
	1-0	R/W	Reserved	
	2	R/W	Old logical screen width bit 8	
	6-3	R/W	Reserved	
	7	R/W	Enable horizontal counter double mode	
3?5	**45**		**Hardware Graphics Cursor Mode (CR45)**	**18-14**
	0	R/W	Enable hardware graphics cursor	
	3-1	R/W	Reserved	
	4	R/W	Set up space at right of bit map for hardware cursor	
	7-5	R/W	Reserved	
3?5	**46–47**		**Hardware Graphics Cursor Origin-X (CR46-CR47)**	**18-14**
	10-0	R/W	X-coordinate of the hardware cursor left side	
	15-11	R/W	Reserved	

Table B-3. Extended CRTC Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	**48–49**		**Hardware Graphics Cursor Origin-Y (CR48-CR49)**	**18-15**
	10-0	R/W	Y-coordinate of the hardware cursor upper line	
	15-11	R/W	Reserved	
3?5	**4A**		**Hardware Graphics Cursor Foreground Stack (CR4A)**	**18-15**
	7-0	R/W	Hardware cursor foreground color (3 registers)	
3?5	**4B**		**Hardware Graphics Cursor Background Stack (CR4B)**	**18-15**
	7-0	R/W	Hardware cursor background color (3 registers)	
3?5	**4C–4D**		**Hardware Graphics Cursor Start Address (CR4C-CR4D)**	**18-16**
	12-0	R/W	Hardware cursor start address	
	15-13	R/W	Reserved	
3?5	**4E**		**Hardware Graphics Cursor Pattern Display Start X-Pixel Position (CR4E)**	**18-16**
	5-0	R/W	Hardware cursor display start x-coordinate	
	7-6	R/W	Reserved	
3?5	**4F**		**Hardware Graphics Cursor Pattern Display Start Y-Pixel Position (CR4F)**	**18-16**
	5-0	R/W	Hardware cursor display start y-coordinate	
	7-6	R/W	Reserved	
3?5	**51**		**Extended System Cont 2 (CR51)**	**18-17**
	1-0	R/W	Old display start address bits 19-18	
	3-2	R/W	Old CPU base address bits 19-18	
	5-4	R/W	Logical screen width bits 9-8	
	6	R/W	Disable split transfers	
	7	R/W	Reserved	
3?5	**52**		**Extended BIOS Flag 1 (CR52)**	**18-18**
	7-0	R/W	Used by the video BIOS	
3?5	**53**		**Extended Memory Cont 1 (CR53)**	**18-18**
	0	R/W	Enable alpha pitching	
	2-1	R/W	Big endian data byte swap for linear addressing	
	4-3	R/W	MMIO type enable and select	
	5	R/W	MMIO window at B8000H	
	6	R/W	Enable nibble swap	
	7	R/W	Reserved	
3?5	**54**		**Extended Memory Cont 2 (CR54)**	**18-19**
	1-0	R/W	Big endian data swap (not linear addressing or image write)	
	7-2	R/W	Reserved	

Table B-3. Extended CRTC Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	55		**Extended DAC Control (CR55)**	18-19
	1-0	R/W	RAMDAC register select RS[3:2]	
	3	R/W	Use external RAMDAC	
	4	R/W	Enable X-11 windows hardware cursor mode	
	6-5	R/W	Reserved	
	7	R/W	VCLK output pin is tri-stated	
3?5	56		**External Sync Cont 1 (CR56)**	18-20
	0	R/W	Reserved	
	1	R/W	HSYNC output buffer tri-stated	
	2	R/W	VSYNC output buffer tri-stated	
	7-3	R/W	Reserved	
3?5	57		**Block Write Control (CR58)**	18-21
	4-0	R/W	Reserved	
	5	R/W	Block write minimum transfer width	
	6	R/W	Reserved	
	7	R/W	Enable block writes	
3?5	58		**Linear Address Window Control (CR58)**	18-21
	1-0	R/W	Linear addressing window size	
	3-2	R/W	Reserved	
	4	R/W	Enable linear addressing	
	5	R/W	Reserved	
	6	R/W	Select 256 word SAM control	
	7	R/W	RAS precharge time increased	
3?5	59-5A		**Linear Address Window Position (CR59-5A)**	18-23
	15-0	R/W	Linear addressing window position bits 31-16	
3?5	5C		**General Out Port (CR5C)**	18-24
	7-0	R/W	General Output Port	
3?5	5D		**Extended Horizontal Overflow (CR5D)**	18-24
	0	R/W	Horizontal total bit 8 (CR0)	
	1	R/W	Horizontal display end bit 8 (CR1)	
	2	R/W	Start horizontal blank bit 8 (CR2)	
	3	R/W	End horizontal blank bit 7 (CR3, CR5)	
	4	R/W	Start horizontal sync position bit 8 (CR4)	
	5	R/W	End horizontal sync position bit 6 (CR5)	
	6	R/W	Data Transfer Position bit 8 (CR3B)	
	7	R/W	Reserved	

Table B-3. Extended CRTC Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	**5E**		**Extended Vertical Overflow (CR5E)**	**18-25**
	0	R/W	Vertical total bit 10 (CR6)	
	1	R/W	Vertical display end bit 10 (CR12)	
	2	R/W	Start vertical blank bit 10 (CR15)	
	3	R/W	Reserved	
	4	R/W	Vertical retrace start bit 10 (CR10)	
	5	R/W	Reserved	
	6	R/W	Line compare position bit 10 (CR18)	
	7	R/W	Reserved	
3?5	**61**		**Extended Memory Control 4 (CR61)**	**18-25**
	4-0	R/W	Reserved	
	6-5	R/W	Big endian data byte swap (image writes)	
	7	R/W	Reserved	
3?5	**63**		**Extended Control (CR63)**	**18-26**
	0	R/W	Enable all accelerated modes	
	1	R/W	Software reset of S3D Engine	
	2	R/W	Reserved	
	3	R/W	Enable PCI disconnects under certain FIFO conditions	
	7-4	R/W	Delay HSYNC/VSYNC by DCLKs	
3?5	**65**		**Extended Miscellaneous Control (CR65)**	**18-27**
	0	R/W	Delay falling edge of SE	
	2-1	R/W	Reserved	
	5-3	R/W	Delay BLANK by DCLK	
	7-6	R/W	Address adjustment for split transfers	
3?5	**66**		**Extended Miscellaneous Control 1 (CR66)**	**18-27**
	2-0	R/W	Divide SC, SE and ICLK	
	3	R/W	Invert SC	
	5-4	R/W	SID mode select	
	6	R/W	PA[15:0] are tri-stated off	
	7	R/W	Enable PCI bus disconnect during burst cycles	
3?5	**67**		**Extended Miscellaneous Control 2 (CR67)**	**18-28**
	0	R/W	ICLK is in phase with DCLK	
	1	R/W	Enable gamma correction	
	3-2	R/W	Select Streams Processor mode	
	7-4	R/W	Select RAMDAC color mode	

Table B-4. Extended CRTC Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3?5	**68**		**Configuration 3 (CR68)**	**18-29**
	0	R/W	CAS, OE trailing edge MSB	
	1	R/W	CAS, OE leading edge MSB	
	2	R/W	RAS low timing select	
	3	R/W	RAS precharge timing select	
	6-4	R/W	BIOS area	
	7	R/W	Reserved	
3?5	**69**		**Extended System Control 3 (CR69)**	**18-30**
	4-0	R/W	Display start address bits 20-16	
	7-5	R/W	Reserved	
3?5	**6A**		**Extended System Control 4 (CR6A)**	**18-31**
	6-0	R/W	CPU base address bits 20-14	
	7	R/W	Reserved	
3?5	**6B**		**Extended BIOS Flag 3 (CR6B)**	**18-31**
	7-0	R/W	Used by the video BIOS	
3?5	**6C**		**Extended BIOS Flag 4 (CR6C)**	**18-31**
	7-0	R/W	Used by the video BIOS	
3?5	**6D**		**Signal Delay (CR6D)**	**18-32**
	3-0	R/W	Delay BLANK by ICLKs	
	7-4	R/W	Delay SC by ICLKs	
3?5	**6E**		**RAMDAC Signature Data (CR6E)**	**18-32**
	7-0	R/W	RAMDAC signature data	
3?5	**6F**		**Configuration 4 (CR6F)**	**18-33**
	0	R/W	Reserved	
	1	R/W	Select I/O address for MMFF20	
	2	R/W	Disable effect of bit 1 of this register	
	3	R/W	WE trailing edge delay MSB	
	4	R/W	WE leading edge delay MSB	
	7-5	R/W	Reserved	
3?5	**80**		**Signal Drive Strength (CR80)**	**18-34**
	0	R/W	Select RAS drive strength	
	1	R/W	Select CAS drive strength	
	2	R/W	Select OE/WE drive strength	
	3	R/W	Select MA drive strength	
	4	R/W	Select PD drive strength	
	5	R/W	Select SC drive strength	
	6	R/W	Select SE drive strength	
	7	R/W	Select DSF drive strength	

B.4 S3d REGISTERS

This section lists the registers which support the S3d Engine functions. All of these registers are enabled only if bit 0 of the System Configuration register (CR40) is set to 1.

Table B-4. Color Pattern Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
			Color Pattern Registers	**19-3**
A100	31-0	R/W	First pattern register	
A104	31-0	R/W	Second pattern register	
.				
.				
.				
A1BC	31-0	R/W	Last pattern register	

Table B-5. S3d 2D Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
AxD4			**Source Base Address**	**19-4**
	2-0	R/W	Reserved = 0	
	21-3	R/W	Source base address	
	31-22	R/W	Reserved	
AxD8			**Destination Base Address**	**19-4**
	2-0	R/W	Reserved = 0	
	21-3	R/W	Destination base address	
	31-22	R/W	Reserved	
AxDC			**Left/Right Clipping**	**19-5**
	10-0	R/W	Left clipping limit	
	15-11	R/W	Reserved	
	26-16	R/W	Right clipping limit	
	31-27	R/W	Reserved	
AxE0			**Top/Bottom Clipping**	**19-6**
	10-0	R/W	Bottom clipping limit	
	15-11	R/W	Reserved	
	26-16	R/W	Top clipping limit	
	31-27	R/W	Reserved	

Table B-5. S3d 2D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
AxE4			**Destination/Source Stride**	**19-6**
	11-0	R/W	Source stride	
	15-12	R/W	Reserved	
	27-16	R/W	Destination stride	
	31-28	R/W	Reserved	
AxE8			**Mono Pattern 0**	**19-7**
	31-0	R/W	Mono pattern 0	
AxEC			**Mono Pattern 1**	**19-7**
	31-0	R/W	Mono pattern 1	
AxF0			**Mono Pattern Background Color**	**19-8**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
AxF4			**Mono Pattern Foreground Color**	**19-9**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
A4F8			**Source Background Color**	**19-10**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
A4FC			**Source Foreground Color**	**19-11**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
Ax00			**Command Set**	**19-12**
	0	R/W	Enable autoexecute	
	1	R/W	Enable hardware clipping	
	4-2	R/W	Destination color format	
	5	R/W	Update screen with new pixel	
	6	R/W	Mono source	
	7	R/W	Image source data from CPU	
	8	R/W	Mono pattern	
	9	R/W	Transparent transfers	
	11-10	R/W	Image transfer alignment	

Table B-5. S3d 2D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
Ax00			**Command Set (continued)**	**19-12**
	13-12	R/W	First doubleword offset for image transfers	
	15-14	R/W	Reserved	
	16	R/W	Disable block writes for this command	
	24-17	R/W	Select one of 256 ROPs	
	25	R/W	X positive BitBLT	
	26	R/W	Y positive BitBLT	
	30-27	R/W	2D command	
	31	R/W	Select 2D or 3D command	
A504			**Rectangle Width/Height**	**19-15**
	10-0	R/W	Rectangle height	
	15-11	R/W	Reserved	
	26-16	R/W	Rectangle width	
	31-27	R/W	Reserved	
A508			**Rectangle Source XY**	**19-16**
	10-0	R/W	Rectangle source Y	
	15-11	R/W	Reserved	
	26-16	R/W	Rectangle source X	
	31-27	R/W	Reserved	
A50C			**Rectangle Destination XY**	**19-16**
	10-0	R/W	Rectangle destination Y	
	15-11	R/W	Reserved	
	26-16	R/W	Rectangle destination X	
	31-27	R/W	Reserved	
A96C			**Line Draw Endpoints**	**19-17**
	15-0	R/W	End 1	
	31-16	R/W	End 2	
A970			**Line Draw X Delta**	**19-18**
	31-0	R/W	X delta	
A974			**Line Draw X Start**	**19-18**
	31-0	R/W	X start	
A978			**Line Draw Y Start**	**19-19**
	10-0	R/W	Y start	
	31-11	R/W	Reserved	

Table B-5. S3d 2D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
A97C			**Line Draw Y Count**	**19-19**
	10-0	R/W	Scan line count	
	30-11	R/W	Reserved	
	31	R/W	Drawing direction from left to right	
AD68			**Polygon Right X Delta**	**19-20**
	31-0	R/W	Right edge X delta	
AD6C			**Polygon Right X Start**	**19-20**
	31-0	R/W	Right edge X start	
AD70			**Polygon Left X Delta**	**19-21**
	31-0	R/W	Left edge X delta	
AD74			**polygon left X Start**	**19-21**
	31-0	R/W	Left edge X start	
AD78			**Polygon Y Start**	**19-22**
	10-0	R/W	Top side of the clipping rectangle	
	31-11	R/W	Reserved	
AD7C			**Polygon Y Count**	**19-22**
	10-0	R/W	Scan line count	
	27-11	R/W	Reserved	
	28	R/W	Update right edge	
	29	R/W	Update left edge	
	31-30	R/W	Reserved	

Table B-6. S3d 3D Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
BxD4			**Z-Buffer Base Address**	**19-24**
	2-0	R/W	Reserved = 0	
	21-3	R/W	Z-buffer base address	
	31-22	R/W	Reserved	
BxD8			**Destination Base Address**	**19-24**
	2-0	R/W	Reserved = 0	
	21-3	R/W	Destination base address	
	31-22	R/W	Reserved	
BxDC			**Left/Right Clipping**	**19-25**
	10-0	R/W	Left clipping limit	
	15-11	R/W	Reserved	
	26-16	R/W	Right clipping limit	
	31-27	R/W	Reserved	
BxE0			**Top/Bottom Clipping**	**19-25**
	10-0	R/W	Bottom clipping limit	
	15-11	R/W	Reserved	
	26-16	R/W	Top clipping limit	
	31-27	R/W	Reserved	
BxE4			**Destination/Source Stride**	**19-26**
	11-0	R/W	Source stride	
	15-12	R/W	Reserved	
	27-16	R/W	Destination stride	
	31-28	R/W	Reserved	
BxE8			**Z-Stride**	**19-26**
	11-0	R/W	Z stride	
	31-12	R/W	Reserved	
BxEC			**Texture Base Address**	**19-27**
	2-0	R/W	Reserved = 0	
	21-3	R/W	Texture base address	
	31-22	R/W	Reserved	
B4F0			**Texture Border Color**	**19-27**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	

Table B-6. S3d 3D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
BxF4			**Fog Color**	**19-28**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
B4F8			**Color0**	**19-29**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
B4FC			**Color1**	**19-30**
	7-0	R/W	Data 1	
	15-8	R/W	Data 2	
	23-16	R/W	Data 3	
	31-24	R/W	Reserved	
Bx00			**Command Set**	**19-31**
	0	R/W	Enable autoexecute	
	1	R/W	Enable hardware clipping	
	4-2	R/W	Destination color format	
	7-5	R/W	Texel color format	
	11-8	R/W	MIPMAP level size	
	14-12	R/W	Texture filtering mode	
	16-15	R/W	Texture blending mode	
	17	R/W	Enable fogging	
	19-18	R/W	Alpha blending control	
	22-20	R/W	Z-buffer compare mode	
	23	R/W	Update z-buffer	
	25-24	R/W	Z-buffering mode	
	26	R/W	Enable texture wrapping	
	30-27	R/W	3D command	
	31	R/W	Select 2D or 3D command	

Table B-6. S3d 3D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
B144			**3D Line Draw GB Delta**	**19-34**
	15-0	R/W	Blue delta	
	31-16	R/W	Green delta	
B148			**3D Line Draw AR Delta**	**19-34**
	15-0	R/W	Red delta	
	31-16	R/W	Alpha delta	
B14C			**3D Line Draw GB Start**	**19-35**
	15-0	R/W	Blue start	
	31-16	R/W	Green start	
B150			**3D Line Draw AR Start**	**19-35**
	15-0	R/W	Red start	
	31-16	R/W	Alpha start	
B158			**3D Line Draw Z Delta**	**19-36**
	31-0	R/W	Z delta	
B15C			**3D Line Draw Z Start**	**19-36**
	31-0	R/W	Z start	
B16C			**3D Line Draw Endpoints**	**19-37**
	15-0	R/W	End 1	
	31-16	R/W	End 2	
B170			**3D Line Draw X Delta**	**19-37**
	31-0	R/W	X delta	
B174			**3D Line Draw X Start**	**19-38**
	31-0	R/W	X start	
B178			**3d Line Draw Y Start**	**19-38**
	10-0	R/W	Y start	
	31-11	R/W	Reserved	
B17C			**3d Line Draw Y Count**	**19-39**
	10-0	R/W	Scan line count	
	30-11	R/W	Reserved	
	31	R/W	Drawing direction is left to right	

Table B-6. S3d 3D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
B504			**Triangle Base V**	**19-39**
	19-0	R/W	Base V	
	31-20	R/W	Reserved	
B508			**Triangle Base U**	**19-40**
	19-0	R/W	Base U	
	31-20	R/W	Reserved	
B50C			**Triangle WX Delta**	**19-40**
	31-0	R/W	WX delta	
B510			**Triangle WY Delta**	**19-41**
	31-0	R/W	WY delta	
B514			**Triangle W Start**	**19-41**
	31-0	R/W	W start	
B518			**Triangle DX Delta**	**19-42**
	31-0	R/W	DX delta	
B51C			**Triangle VX Delta**	**19-42**
	31-0	R/W	VX delta	
B520			**Triangle UX Delta**	**19-43**
	31-0	R/W	UX delta	
B524			**Triangle DY Delta**	**19-43**
	31-0	R/W	DY delta	
B528			**Triangle VY Delta**	**19-44**
	31-0	R/W	VY delta	
B52C			**Triangle UY Delta**	**19-44**
	31-0	R/W	UY delta	
B530			**Triangle D Start**	**19-45**
	31-0	R/W	D start	
B534			**Triangle V Start**	**19-45**
	31-0	R/W	V start	
B538			**Triangle U Start**	**19-46**
	31-0	R/W	U start	
B53C			**Triangle GBX Delta**	**19-46**
	15-0	R/W	Blue X delta	
	31-16	R/W	Green X delta	
B540			**Triangle ARX Delta**	**19-47**
	15-0	R/W	Red X delta	
	31-16	R/W	Alpha X delta	

Table B-6. S3d 3D Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
B544			**Triangle GBY Delta**	**19-47**
	15-0	R/W	Blue Y delta	
	31-16	R/W	Green Y delta	
B548			**Triangle ARY Delta**	**19-48**
	15-0	R/W	Red Y delta	
	31-16	R/W	Alpha Y delta	
B54C			**Triangle GB Start**	**19-48**
	15-0	R/W	Blue start	
	31-16	R/W	Green start	
B550			**Triangle AR Start**	**19-49**
	15-0	R/W	Red start	
	31-16	R/W	Alpha start	
B554			**Triangle ZX Delta**	**19-49**
	31-0	R/W	ZX delta	
B558			**Triangle ZY Delta**	**19-50**
	31-0	R/W	ZY delta	
B55C			**Triangle Z Start**	**19-50**
	31-0	R/W	Z start	
B560			**Triangle XY12 Delta**	**19-51**
	31-0	R/W	XY12 delta	
B564			**Triangle X12 End**	**19-51**
	31-0	R/W	X12 end	
B568			**Triangle XY01 Delta**	**19-52**
	31-0	R/W	XY01 delta	
B56C			**Triangle X01 End**	**19-52**
	31-0	R/W	X01 end	
B570			**Triangle XY02 Delta**	**19-53**
	31-0	R/W	XY02 delta	
B574			**Triangle X Start**	**19-53**
	31-0	R/W	X start	
B578			**Triangle Y Start**	**19-54**
	31-0	R/W	Y start	
B57C			**Triangle Y Count**	**19-54**
	10-0	R/W	Scan line count 12	
	15-11	R/W	Reserved	
	26-16	R/W	Scan line count 01	
	30-27	R/W	Reserved	
	31	R/W	Render the triangle from right to left	

B.5 STREAMS PROCESSOR REGISTERS

Table B-7. Streams Processor Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
8180			**Primary Stream Control**	**20-2**
	23-0	R/W	Reserved	
	26-24	R/W	Primary stream input data format	
	27	R/W	Reserved	
	30-28	R/W	Primary stream filter characteristics	
	31	R/W	Reserved	
8184			**Color/Chroma Key Control**	**20-3**
	7-0	R/W	B/V/Cr key value (lower bound for chroma)	
	15-8	R/W	G/U/Cb key value (lower bound for chroma)	
	23-16	R/W	R/Y key value (lower bound for chroma)	
	26-24	R/W	RGB color comparison precision	
	27	R/W	Reserved	
	28	R/W	Color key control (full compare or bit 16 of 1.5.5.5)	
	31-29	R/W	Reserved	
8190			**Secondary Stream Control**	**20-4**
	12-0	R/W	DDA horizontal accumulator initial value	
	23-13	R/W	Reserved	
	26-24	R/W	Secondary stream input data format	
	27	R/W	Reserved	
	30-28	R/W	Secondary stream filter characteristics	
	31	R/W	Reserved	
8194			**Chroma Key Upper Bound**	**20-5**
	7-0	R/W	V/Cr key value (upper bound)	
	15-8	R/W	U/Cb key value (upper bound)	
	23-16	R/W	Y key value (upper bound)	
	31-24	R/W	Reserved	
8198			**Secondary Stream Stretch/Filter Constants**	**20-5**
	11-0	R/W	K1 horizontal scale factor	
	15-12	R/W	Reserved	
	27-16	R/W	K2 horizontal scale factor	
	31-28	R/W	Reserved	

Table B-7. Streams Processor Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
81A0			**Blend Control**	**20-6**
	1-0	R/W	Reserved	
	4-2	R/W	Secondary stream blend coefficient	
	9-5	R/W	Reserved	
	12-10	R/W	Primary stream blend coefficient	
	23-13	R/W	Reserved	
	26-24	R/W	Compose mode	
	31-27	R/W	Reserved	
81C0			**Primary Stream Frame Buffer Address 0**	**20-7**
	21-0	R/W	Primary stream frame buffer starting address 0	
	31-22	R/W	Reserved	
81C4			**Primary Stream Frame Buffer Address 1**	**20-7**
	21-0	R/W	Primary stream frame buffer starting address 1	
	31-22	R/W	Reserved	
81C8			**Primary Stream Stride**	**20-8**
	11-0	R/W	Primary stream stride	
	31-12	R/W	Reserved	
81CC			**Double Buffer/LPB Support**	**20-8**
	0	R/W	Select primary frame buffer address 1	
	2-1	R/W	Select secondary frame buffer address	
	3	R/W	Reserved	
	4	R/W	Select LPB frame buffer start address 1	
	5	R/W	LPB input buffer select loading at end of frame	
	6	R//w	Selected LPB input buffer toggles at end of frame	
	31-7	R/W	Reserved	
81D0			**Secondary Stream Frame Buffer Address 0**	**20-10**
	21-0	R/W	Secondary stream frame buffer starting address 0	
	31-22	R/W	Reserved	
81D4			**Secondary Stream Frame Buffer Address 1**	**20-10**
	21-0	R/W	Secondary stream frame buffer starting address 1	
	31-22	R/W	Reserved	
81D8			**Secondary Stream Stride**	**20-11**
	11-0	R/W	Secondary stream stride	
	31-12	R/W	Reserved	

Table B-7. Streams Processor Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
81DC			**Blend Control**	**20-11**
	2-0	R/W	Reserved	
	12-3	R/W	Pixel stop fetch position	
	18-13	R/W	Reserved	
	28-19	R/W	Pixel start fetch position	
	29	R/W	Reserved	
	30	R/W	Primary stream on top	
	31	R/W	Enable opaque overlay control	
81E0			**K1 Vertical Scale Factor**	**20-12**
	11-0	R/W	K1 vertical scale factor	
	31-12	R/W	Reserved	
81E4			**K2 Vertical Scale Factor**	**20-13**
	11-0	R/W	K2 vertical scale factor	
	31-12	R/W	Reserved	
81E8			**DDA Vertical Accumulator Initial Value**	**20-13**
	12-0	R/W	DDA vertical accumulator initial value	
	14-13	R/W	Reserved	
	15	R/W	Enable vertical filtering	
	31-13	R/W	Reserved	
81EC			**Streams FIFO Control**	**20-14**
	17-0	R/W	Reserved	
	18	R/W	Specify memory size for LPB memory cycles	
	19	R/W	Do not tri-state PD[63:16] during ROM cycles	
	20	R/W	Disable memory arbitration for ROM cycles	
	21	R/W	Do not delay PD output	
	31-22	R/W	Reserved	
81F0			**Primary Stream Window Start Coordinates**	**20-15**
	10-0	R/W	Primary stream Y start	
	15-11	R/W	Reserved	
	26-16	R/W	Primary stream X start	
	31-27	R/W	Reserved	
81F4			**Primary Stream Window Size**	**20-15**
	10-0	R/W	Primary stream height	
	15-11	R/W	Reserved	
	26-16	R/W	Primary stream width	
	31-27	R/W	Reserved	

Table B-7. Streams Processor Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
81F8			**Secondary Stream Window Start Coordinates**	**20-16**
	10-0	R/W	Secondary stream Y start	
	15-11	R/W	Reserved	
	26-16	R/W	Secondary stream X start	
	31-27	R/W	Reserved	
81FC			**Secondary Stream Window Size**	**20-16**
	10-0	R/W	Secondary stream height	
	15-11	R/W	Reserved	
	26-16	R/W	Secondary stream width	
	31-27	R/W	Reserved	

B.6 MEMORY PORT CONTROLLER REGISTERS

Table B-8. Memory Port Controller Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
8200			**FIFO Control**	**21-1**
	4-0	R/W	Primary/secondary stream FIFO boundary	
	5	R/W	Reserved	
	10-6	R/W	Secondary stream threshold	
	11	R/W	Reserved	
	16-12	R/W	Primary stream threshold	
	31-17	R/W	Reserved	
8204			**MIU Control**	**21-2**
	0	R/W	RAS inactive at end of cycle or start of next cycle	
	1	R/W	RAS pre-charge = 1.5 MCLKs	
	2	R/W	RAS low time = 2.5 MCLKs	
	6-3	R/W	Reserved	
	7	R/W	CAS/OE timing adjustment	
	8	R/W	WE delay	
	9	R/W	Stream Processor low priority request servicing	
	10	R/W	Delay RAS rising edge by 1/2 MCLK	
	31-11	R/W	Reserved	
8208			**Streams Timeout**	**21-3**
	7-0	R/W	Secondary stream timeout	
	15-8	R/W	Primary stream timeout	
	16	R/W	Secondary stream wins memory arbitration in case of a tie	
	31-17	R/W	Reserved	
820C			**Miscellaneous Timeout**	**21-4**
	7-0	R/W	CPU timeout	
	15-8	R/W	S3D Engine Timeout	
	23-16	R/W	LPB Timeout	
	31-24	R/W	External memory master timeout	
8220			**DMA Read Base Address**	**21-5**
	2-0	R/W	Reserved = 0	
	22-3	R/W	DMA read base address	
	31-23	R/W	Reserved	

Table B-8. Memory Port Controller Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
8224			**DMA Read Stride/Width**	**21-5**
	1-0	R/W	Address tiling control	
	2	R/W	Reserved = 0	
	11-3	R/W	DMA read stride	
	15-12	R/W	Reserved	
	18-16	R/W	Reserved = 0	
	27-19	R/W	DMA read width	
	31-28	R/W	Reserved	

B.7 MISCELLANEOUS REGISTERS

Table B-9. Miscellaneous Registers

Add dress	Index Bit(s)	R/W	Register Name Bit Description	Description Page
8504			**Subsystem Status**	**22-1**
	0	R	Vertical sync interrupt generated	
	1	R	S3D Engine interrupt generated	
	2	R	Command FIFO overflow interrupt generated	
	3	R	Command FIFO empty interrupt generated	
	4	R	Host DMA done interrupt generated	
	5	R	Command DMA done interrupt generated	
	6	R	S3D FIFO empty interrupt generated	
	7	R	LPB interrupt generated	
	12-8	R	S3D FIFO slots free	
	13	R	S3D Engine idle	
	31-14	R	Reserved	
8504			**Subsystem Control**	**22-2**
	0	W	Vertical sync interrupt cleared	
	1	W	S3D Engine interrupt cleared	
	2	W	Command FIFO overflow interrupt cleared	
	3	W	Command FIFO empty interrupt cleared	
	4	W	Host DMA done interrupt cleared	
	5	W	Command DMA done interrupt cleared	
	6	W	S3D FIFO empty interrupt cleared	
	7	W	Host DMA done interrupt enabled	
	8	W	Vertical sync interrupt enabled	
	9	W	S3D Engine interrupt enabled	
	10	W	Command FIFO overflow interrupt enabled	
	11	W	Command FIFO empty interrupt enabled	
	12	W	Command DMA done interrupt enabled	
	13	W	S3D FIFO empty interrupt enabled	
	15-14	W	S3D Engine software reset select	
	31-16	W	Reserved	
850C			**Advanced Function Control**	**22-4**
	0	R/W	Enable accelerated modes (enhanced and VESA non-planar)	
	3-1	R/W	Reserved	
	4	R/W	Enable linear addressing	
	5	R/W	Reserved	
	9-6	R	BIU FIFO Status	
	31-10	R/W	Reserved	

B.8 DMA REGISTERS

Table B-10. DMA Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
8580				**23-2**
	0	R/W	Reserved	
	1	R/W	Video DMA write	
	31-2	R/W	DMA starting memory address	
8584			**Video DMA Transfer Length**	**23-3**
	1-0	R/W	Reserved	
	23-2	R/W	DMA transfer length	
	31-24	R/W	Reserved	
8588			**Video DMA Transfer Length**	**23-3**
	0	R/W	Enable video/graphics DMA	
	31-1	R/W	Reserved	
8590			**Command DMA Base Address**	**23-4**
	0	R/W	Reserved	
	1	R/W	Specify 4/64 KByte buffer size	
	11-2	R/W	Reserved	
	31-12	R/W	Command DMA buffer base address	
8594			**Command DMA Write Pointer**	**23-5**
	1-0	RW	Reserved	
	15-2	R/W	Write pointer	
	16	R/W	Write pointer updated	
	31-17	R/W	Reserved	
8598			**DMA Read Pointer**	**23-5**
	1-0	R/W	Reserved	
	15-2	R/W	Read pointer	
	31-16	R/W	Reserved	
859C			**Command DMA Enable**	**23-6**
	0	RW	Enable Command DMA	
	31-1	R/W	Reserved	

B.9 LPB REGISTERS

Table B-11. LPB Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
FF00			**LPB Mode**	**24-1**
	0	R/W	Enable LPB	
	3-1	R/W	LPB mode	
	4	R/W	Reset LPB	
	5	R/W	Write every other received frame to memory	
	6	R/W	No byte swap for incoming video	
	8-7	R/W	Reserved	
	9	R/W	LPB vertical sync is active high	
	10	R/W	LPB horizontal sync is active high	
	11	W	CPU VSYNC	
	12	W	CPU HSYNC	
	13	W	Load base address currently pointed to	
	15-14	R/W	Reserved	
	17-16	R/W	Maximum compressed data bust size	
	20-18	R/W	Reserved	
	22-21	R/W	Video FIFO threshold	
	23	R/W	Reserved	
	24	R/W	LPB clock driven by LCLK	
	25	R/W	Don't add stride after first HSYNC	
	26	R/W	Invert the LCLK input	
	31-27	R/W	Reserved	

Table B-11. LPB Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
FF04			**LPB FIFO Status**	**24-4**
	3-0	R	LPB output FIFO status	
	10-4	R	Reserved	
	11	R	LPB output FIFO full	
	12	R	LPB output FIFO empty	
	13	R	LPB output FIFO almost empty	
	19-14	R	Reserved	
	20	R	LPB video FIFO 0 full	
	21	R	LPB video FIFO 0 empty	
	22	R	LPB video FIFO 0 almost empty	
	28-23	R	Reserved	
	29	R	LPB video FIFO 1 full	
	30	R	LPB video FIFO 1 empty	
	31	R	LPB video FIFO 1 almost empty	
FF08			**LPB Interrupt Flags**	**24-5**
	0	R/W	LPB Output FIFO empty	
	1	R/W	HSYNC (end of line) input on pin B5	
	2	R/W	VSYNC (end of frame) input on pin A5	
	3	R/W	Serial port start condition detected	
	15-4	R/W	Reserved	
	16	R/W	Enable LPB output FIFO empty interrupt	
	17	R/W	Enable HSYNC (end of line) input interrupt	
	18	R/W	Enable VSYNC (end of frame) input interrupt	
	19	R/W	Enable serial port start condition detect interrupt	
	23-20	R/W	Reserved	
	24	R/W	Drive SPCLK low on receipt of a serial port start condition	
	31-25	R/W	Reserved	
FF0C			**LPB Frame Buffer Address 0**	**24-6**
	21-0	R/W	LPB frame buffer address 0	
	31-22	R/W	Reserved	
FF10			**LPB Frame Buffer Address 1**	**24-7**
	21-0	R/W	LPB frame buffer address 1	
	31-22	R/W	Reserved	

Table B-11. LPB Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
FF14			**LPB Direct Read/Write Address**	**24-7**
	20-0	R/W	Address of MPEG decoder address to read/write	
	23-21	R/W	MPEG decoder transaction type	
	31-24	R/W	Reserved	
FF18			**LPB Direct Read/Write Data**	**24-8**
	31-0	R/W	LPB direct read/write data	
FF1C			**LPB General Purpose Input/Output Port**	**24-8**
	3-0	R/W	General purpose output data port	
	7-4	R	General purpose input data port	
	31-8	R/W	Reserved	
FF20			**Serial Port**	**24-9**
	0	R/W	0 = Serial clock write on pin C6, 1 = pin C6 tri-state	
	1	R/W	0 = Serial data write on pin B4, 1 = pin B4 tri-state	
	2	R	0 = Serial clock low on pin C6, 1 = pin C6 tri-state	
	3	R	0 = Serial data low on pin B4, 1 = pin B4 tri-state	
	4	R/W	Enable serial port function	
	5-7	R/W	Reserved	
	8	R	Bit 0 mirror (data on byte lane 2 at E2H)	
	9	R	Bit 1 mirror (data on byte lane 2 at E2H)	
	10	R	Bit 2 mirror (data on byte lane 2 at E2H)	
	11	R	Bit 3 mirror (data on byte lane 2 at E2H)	
	12	R	Bit 4 mirror (data on byte lane 2 at E2H)	
	31-13	R/W	Reserved	
FF24			**LPB Video Input Window Size**	**24-10**
	11-0	R/W	Video input line width	
	15-12	R/W	Reserved	
	24-16	R/W	Video input window height	
	31-25	R/W	Reserved	
FF28			**LPB Video Data Offsets**	**24-11**
	11-0	R/W	Horizontal video data offset	
	15-12	R/W	Reserved	
	24-16	R/W	Vertical video data offset	
	31-25	R/W	Reserved	
FF2C			**LPB Horizontal Decimation Control**	**24-11**
	31-0	R/W	Video data luma mask	
FF30			**LPB Vertical Decimation Control**	**24-12**
	7-0	R	Video data line mask	

Table B-11. LPB Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
FF34			**LPB Line Stride**	**24-12**
	11-0	R/W	Line stride	
	31-12	R/W	Reserved	
FF40			**LPB Output FIFO**	**24-13**
	31-0	R/W	Output FIFO data	

B.10 PCI CONFIGURATION SPACE REGISTERS

When a PCI configuration read or write command is issued, AD[7:0] contain the address of the register in the configuration space to be accessed.

Table B-12. PCI Configuration Space Registers

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
00			**Vendor ID**	**25-1**
	15-0	R	Hardwired to 5333H	
02			**Device ID**	**25-2**
	15-0	R	Hardwired to 5631H (initial stepping)	
04			**Command**	**25-2**
	0	R/W	Response to I/O space accesses enabled	
	1	R/W	Response to memory space accesses enabled	
	2	R/W	Enable bus master operation	
	4-3	R/W	Reserved	
	5	R/W	Enable DAC snooping	
	15-6	R/W	Reserved	
06			**Status**	**25-3**
	8-0	R/W	Reserved	
	10-9	R/W	Hardwired to select medium device select timing	
	11	R/W	Reserved	
	12	R/W	Bus master transaction terminated with target-abort	
	13	R/W	Bus master transaction terminated with master-abort	
	15-14	R/W	Reserved	
08			**Class Code**	**25-3**
	31-0	R	Hardwired to indicate VGA-compatible display controller	
0D			**Latency Timer**	**25-4**
	7-0	R/W	Reserved	
	10-8	R/W	Reserved = 0 (3 LSBs of latency timer)	
	15-11	R/W	Bus master latency timer	
10			**Base Address 0**	**25-4**
	0	R/W	Hardwired to indicate base registers map into memory space	
	2-1	R/W	Hardwired to allow mapping anywhere in 32-bit address space	
	3	R/W	Hardwired to indicate does not meet prefetchable requirements	
	22-4	R/W	Reserved	
	31-23	R/W	Base address 0	
30			**BIOS Base Address**	**25-5**
	0	R/W	Enable access to BIOS ROM address space	
	15-1	R/W	Reserved	
	31-16	R/W	Upper 16 bits of BIOS ROM address	

Table B-12. PCI Configuration Space Registers (continued)

Address	Index Bit(s)	R/W	Register Name Bit Description	Description Page
3C			**Interrupt Line**	**25-3**
	7-0	R/W	Interrupt line routing information	
3D			**Interrupt Pin**	**25-6**
	7-0	R	Hardwired to specify use of INTA	
3E			**Latency/Grant**	**25-6**
	7-0	R/W	Minimum grant	
	15-8	R/W	Maximum latency	

Index

D

E

CORPORATE OFFICES

S3 Incorporated Headquarters
408-980-5400
408-980-7791 (FAX)
www.s3.com (World Wide Web)

S3 Incorporated Asia Pacific
852-2-831-7000
852-2-893-2857 (FAX)

S3 Incorporated Boston
617-242-7707
617-242-7709 (FAX)

S3 Incorporated Florida
305-476-6927
305-476-0945 (FAX)

S3 Incorporated Georgia
770-729-6933
770-729-6931 (FAX)

S3 Incorporated India
91-80-555-0888
91-80-555-0718 (FAX))

S3 Incorporated Japan
81-3-5322-2065
81-3-5322-2066 (FAX)

S3 Incorporated Taiwan
886-2-757-6768
882-2-757-6880 (FAX)

S3 Incorporated Texas
214-479-9500
214-479-9503 (FAX)

U.S. REPRESENTATIVES

CALIFORNIA (NORTHERN)
Trinity Technologies
408-733-9000
408-733-9970 (FAX)

MASSACHUSETTS
Compass Technology
617-272-9990
617-272-9992 (FAX)

MICHIGAN
MicroTech Sales Inc.
313-459-0200
313-459-0232 (FAX)

MINNESOTA
Gleason Associates Inc.
612-447-7020
612-447-7022 (FAX)

NEW JERSEY
Tritek Sales Inc.
609-667-0200
609-667-8741 (FAX)

TEXAS (AUSTIN)
C3 Sales Inc.
512-338-1333
512-338-9185 (FAX)

TEXAS (DALLAS)
C3 Sales Inc.
214-733-0306
214-733-0307 (FAX)

TEXAS (HOUSTON)
C3 Sales Inc.
713-890-3733
713-890-7717 (FAX)

INTERNATIONAL REPRESENTATIVES

CANADA
Peak Marketing
514-696-9298
514-620-9621 (FAX)

CHINA
Fujian Joining Enterprises
86-591-754-4482
86-591-754-4485 (FAX)

FRANCE
Tekelec Airtronic S.A
011-33-1-46-23-24-25
011-33-1-45-07-21-91 (FAX)

GERMANY
Tekelec Airtronic GmbH
011-49-89-51-64-173
011-49-89-51-64-110 (FAX)

ITALY
Acsis S.r.l.
011-39-2-4802-2522
011-39-2-4801-2289 (FAX)

JAPAN
Innotech Corp.
81-45-474-9037
81-45-474-9065 (FAX)

KOREA
Damon Electronics Co. LTD
82-2-588-5353
82-2-588-1085 (FAX)

Phil-Tek Corporation
82-2-532-4125
82-2-534-5460 (FAX)

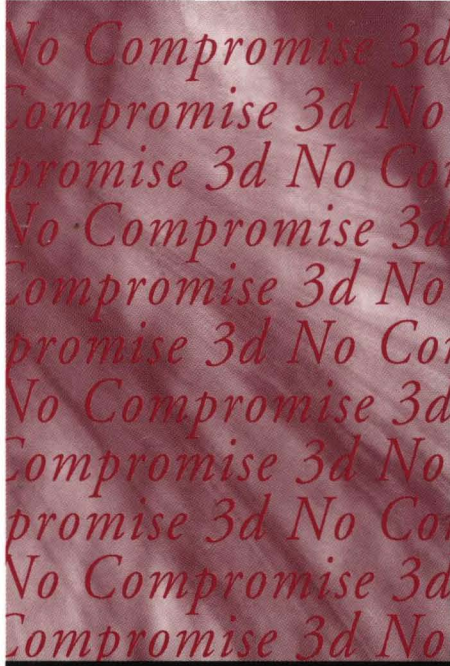
SINGAPORE
Electronic Resources LTD
65-298-0888
65-298-1111 (FAX)

TAIWAN
Synnex Technology Int'l Corp.
886-2-506-3320
886-2-504-8081 (FAX)

Promate Electronic Co. Ltd
886-2-659-0303
822-2-658-0988 (FAX)

UNITED KINGDOM
Manhattan Skyline Ltd
011-44-628-778686
011-44-628-782812 (FAX)

S3 Incorporated, P.O. Box 58058, Santa Clara, CA 95052-8058 Tel: 408-980-5400, Fax: 408-980-5444



S3 Incorporated

P.O. Box 58058

2770 San Tomas Expwy.

Santa Clara, CA 95052-8058

Tel: (408) 980-5400

Fax: (408) 980-5444

http://www.s3.com

Printed in U.S.A. on recycled paper
DB025-A