Monochrome
VME-Graphics Subsystem
Hardware Reference Manual

INTEGRATED SOLUTIONS
1140 Ringwood Court
San Jose, California 95131
(408) 943-1902

490025 Rev. B

October 1985

PREFACE

This manual describes the features, architecture, specifications, configuration, and software interface of the Integrated Solutions Monochrome VME-Graphics Subsystem. The manual is divided into four sections and one appendix:

SECTION 1 describes the general features and architecture of the Monochrome VME-Graphics Subsystem.

SECTION 2 lists the Monochrome VME-Graphics Subsystem specifications.

SECTION 3 provides information regarding the Monochrome VME-Graphics Subsystem configuration.

SECTION 4 describes the Monochrome VME-Graphics Subsystem software interface.

APPENDIX A provides the J1 connector pin assignments for Display Controller boards prior to Revision D.

TABLE OF CONTENTS

SECTION 1: INTRODUCTION 1-1
1.1 Features 1-1
1.2 Architecture 1-2
1.2.1 Display Controller 1-2
1.2.2 Display Memory 1-5

SECTION 2: SPECIFICATIONS 2-1
2.1 Form Factor 2-1
2.2 Power Requirements 2-1
2.3 Environmental 2-1
2.4 System Bus 2-1
2.5 Display Memory Bus 2-4
2.6 Video Interface 2-7
2.7 Interrupt Vector 2-7
2.8 Display Memory 2-8
2.9 Addressing 2-8
2.10 Indicators 2-8

SECTION 3: CONFIGURATION 3-1
3.1 Display Controller Board Jumpers 3-1
3.1.1 Address Map (E1-E6) 3-1
3.1.2 Interrupt Request Level (E7-E10) 3-4
3.1.3 Sequencer Output Control (E11-E14) 3-4
3.1.4 Interrupt Vector (E23-E36) 3-4
3.1.5 ALU Clock Generation (E19-E22, E49, and E50) 3-4
3.1.6 Jumpers (E15-E18) 3-5
3.1.7 Video Output (E37-E42, E47, and E48) 3-5
3.1.8 Jumpers E43-E46 3-5
3.1.9 Vertical and Horizontal SYNC (E53-E56) 3-5
3.2 Display Memory Board Jumpers 3-5
3.2.1 Memory Plane Select (E1-E8) 3-5
3.2.2 Video Latch Enable (E9-E16) 3-5
3.2.3 Address Strobe (E17-E24) 3-5

SECTION 4: SOFTWARE INTERFACE 4-1
4.1 Display Memory 4-1
4.2 Control Registers 4-1
4.2.1 FIFO Register 4-2
4.2.2 Reset Register 4-2
4.3 Microcode Instruction Set 4-2

APPENDIX A: DISPLAY CONTROLLER J1 PIN ASSIGNMENTS A-1

LIST OF FIGURES

Figure 1-1. VME-Graphics Subsystem Block Diagram 1-1

Figure 1-2. Display Controller Block Diagram 1-3

Figure 1-3. Display Memory Block Diagram 1-6

Figure 2-1. VME-Graphics Subsystem Connectors 2-4

Figure 3-1. Display Controller Board (Rev. C and earlier) 3-2

Figure 3-2. Display Controller Board (Rev. D and later) 3-3

Figure 3-3. Interrupt Vector Jumpers 3-4

Figure 3-4. Display Memory Board (Prior to Rev. B) 3-6

LIST OF TABLES

Table 2-1. Power Requirements 2-1

Table 2-2. Display Controller Connector P1 Pin Assignments 2-2

Table 2-3. Display Controller Connector P2 Pin Assignments 2-3

Table 2-4. Display Memory Bus Connector J1 Pin Assignments 2-5

Table 2-5. Display Memory Bus Connector J2 Pin Assignments 2-6

Table 2-6. Video Interface Timing 2-7

Table 3-1. Base Address Jumpers 3-1

Table 3-2. Interrupt Request Levels 3-4

Table 3-3. Color or Monochrome Jumper Configurations 3-5

Table 4-1. Display Memory Address Map 4-1

Table 4-2. Microcode Instruction OpCodes 4-4

Table A-1. Connector J1 Pin Assignments A-1

SECTION 1: INTRODUCTION

1.1 Features

The Monochrome VME-Graphics Subsystem is a high-speed graphics subsystem that supports CRT window management and general screen drawing capability in an industry standard VME bus environment. The VME-Graphics Subsystem is composed of a display controller board and a dual-ported display memory board (see Figure 1-1). The display controller board contains a microcoded graphics processor that executes code to draw and manipulate graphics primitives and the video logic to refresh the display. The graphics processor handles such operations as vector generation, area fill, and bit block transfer, freeing the host CPU to manage other tasks. This results in a more responsive environment when running applications involving both graphics and computational loads.

General features of the VME-Graphics Subsystem include the following:

- Area fill rate — 51 million pixels per second
- Bit block transfer — 21 million pixels per second
- Vector drawing rate — 0.3 million pixels per second
- Total display memory — 512 Kbytes
  - Displayable memory size — 1280 x 1024 bits (two frame buffers)
  - 192 Kbytes of memory available for storing menus, fonts, and display tables

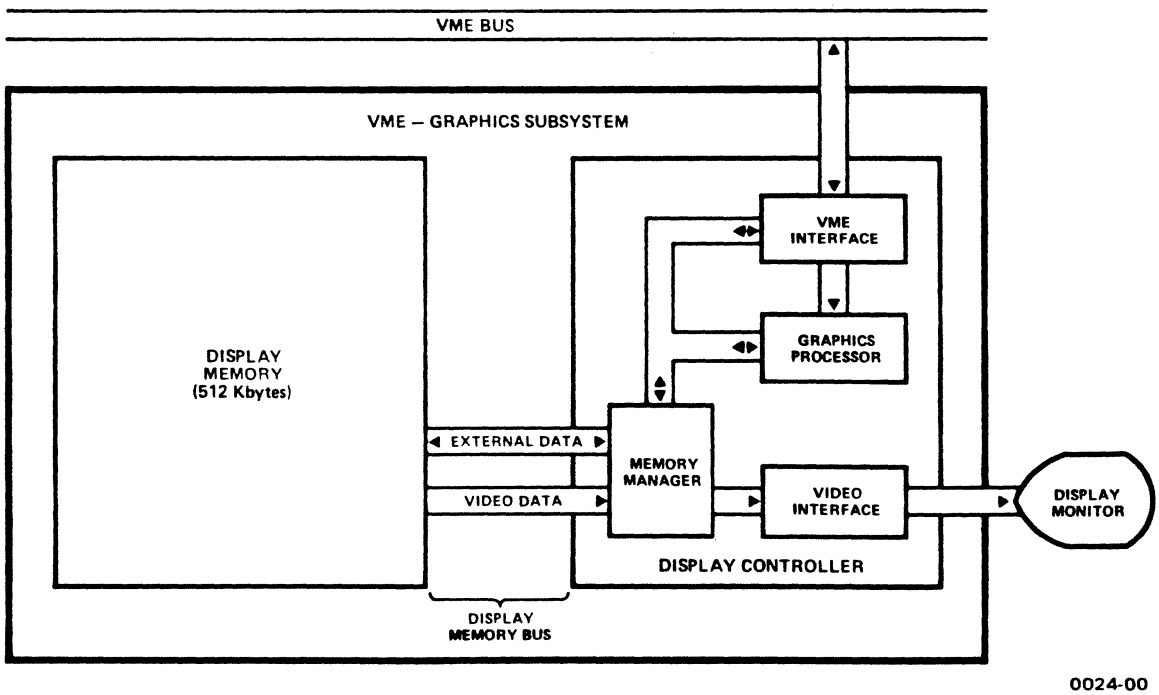

Figure 1-1. VME-Graphics Subsystem Block Diagram

1.2 Architecture

The VME-Graphics Subsystem logic is implemented on two VME double-wide printed circuit boards: the display controller board and the display memory board.

1.2.1 Display Controller

The display controller logic is composed of four major functional units: the graphics processor, video logic, display memory access manager, and VME interface logic (see Figure 1-2). Each of these major functional units is in turn made up of smaller functional blocks which are discussed in the paragraphs that follow.

1.2.1.1 Graphics Processor

As illustrated in Figure 1-2, commands and operands are supplied to the graphics processor from the VME bus through a 64-word deep first-in-first-out register (FIFO). The command FIFO is a Read/Write register to which the host CPU writes a stack of commands to be executed by the graphics processor. The graphics processor then proceeds to execute these commands with no further need of CPU intervention. When execution of the command stack is complete, FIFO empty status is given to the host CPU in one of two ways. If the host CPU issued an Interrupt Enable command, the graphics processor sends an interrupt to the host on a FIFO empty condition. If interrupts are not enabled, the host CPU polls for a *FIFO Empty* status bit. Upon receipt of FIFO Empty status, the CPU can then write another group of commands to the FIFO.

The commands written to the FIFO are processed through the command map PROM which supplies microroutine starting addresses to the microprogram sequencer. The sequencer begins the microinstruction execution sequence of a collection of microcoded routines that are resident in control storage.

Central to the graphics processor is a microcoded arithmetic logic unit (ALU) that resides on the VME bus as a slave co-processing function. The ALU is implemented using a high performance, bipolar, AMD 29116 microprocessor that is optimized specifically for bit-manipulation operations. The 29116 supports a 100ns cycle time providing ten million instructions per second. The ALU also contains a 16-bit barrel shifter which enables the graphics processor to rotate a word any number of bits and merge it with another word in one microcycle.

As peripherals of the ALU, the bit map port and address generators (destination and source) supply the means for communication between the graphics processor and display memory. The bit map port operates as a high performance multiplexing function between the 16-bit painting engine data bus (PD bus) and the 32-bit external data bus (ED bus). The address generators receive input from the ALU and generate the appropriate display memory addresses.

Dual address generators, that is, separate source and destination address generators, increase the execution speed of certain graphics primitives. For example, the box copy (boxc) command copies a rectangular region from one part of the display area to another. The instruction parameters include the existing box's coordinates (upper left corner x,y coordinates), width, and height, as well as the coordinates of the new (destination) location. These instruction parameters are processed and result in source address and destination address generation. Each address is incremented under microcode control after each 64-bit data transfer. Consequently, the source and destination addresses for each data transfer are available simultaneously. This minimizes the number of instruction cycles required to complete the boxc operation.

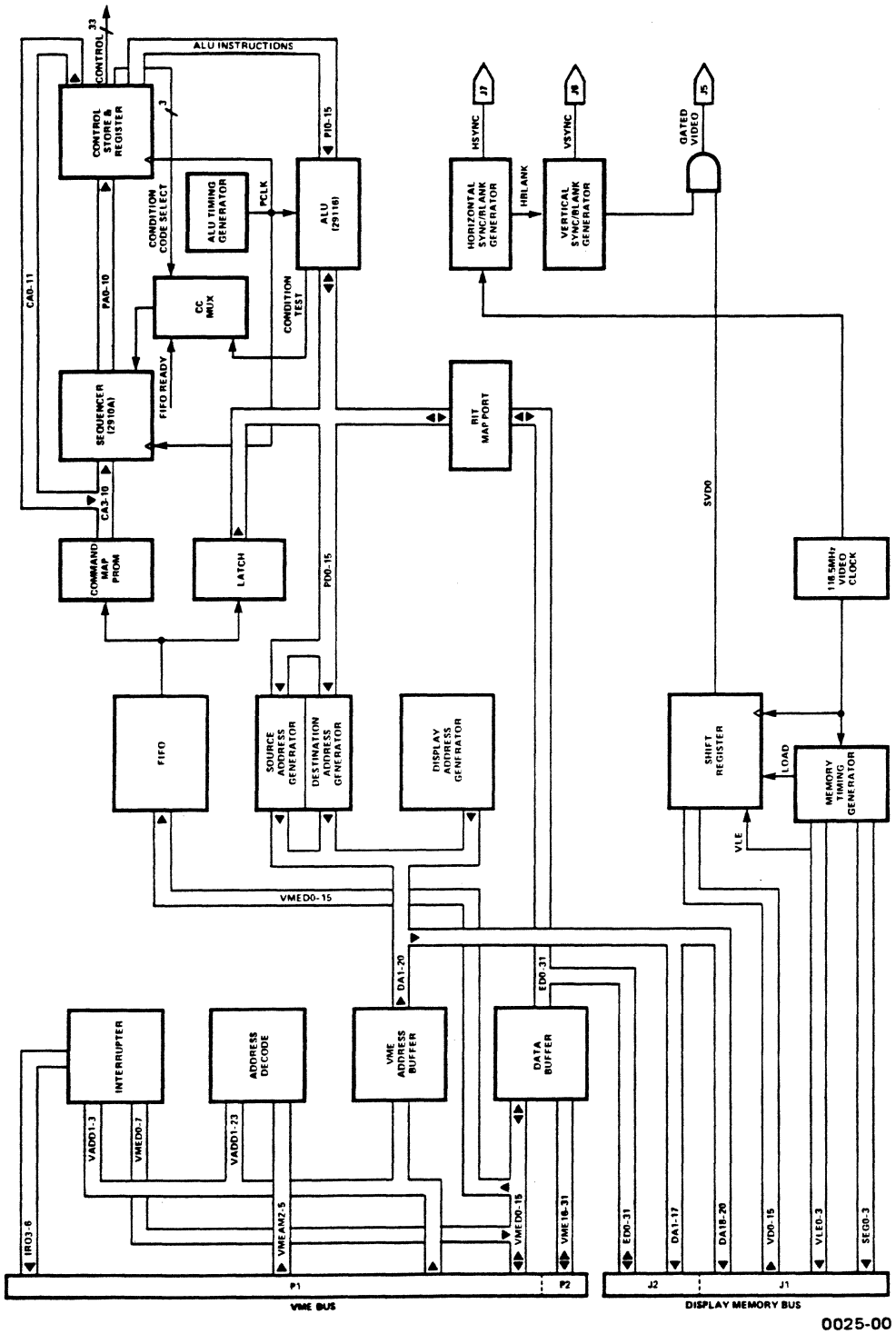

Figure 1-2. Display Controller Block Diagram

1.2.1.2 Video Logic

The Video Logic resident on the display controller board requests a memory cycle from the Memory Access Manager every 550 nanoseconds and retrieves 64 bits of data during its memory cycle. This data is serialized and converted to the emitter-coupled-logic (ECL) levels appropriate for driving the high bandwidth monochrome monitor (via connector J5).

Generation of the video timing signals (horizontal and vertical sync) is also a function of the video logic. HSYNC and VSYNC are sent separately to the monitor via display controller board connectors J7 and J6, respectively.

1.2.1.3 Display Memory Access Manager

The display memory access manager arbitrates access to display memory between the refresh logic and the external port. The display memory access manager uses the 116.5 MHz video clock to generate the basic timing signals for the display memory (see Figure 1-2). The normal memory cycle is 550 nanoseconds with access divided equally between the display refresh and the external port. This memory interleaving allows an external source (either the VME bus or the graphics processor) to have access to the display memory during the external port period. External access is asynchronous with the display refresh. External access time varies from 275 to 550 nanoseconds.

Access to display memory over the external port is shared by the graphics processor and the VME bus. Arbitration here is under microcode control. The host processor may access the display memory from the VME bus any time that the graphics processor is in an idle loop waiting to receive commands from the host CPU. Once the graphics processor receives a command, VME bus accesses are no longer recognized. Any attempted display memory access during this time will result in a bus time-out error since DTACK will not be asserted.

This shared access to the display memory enables the host CPU to download image data files, menus, fonts, and graphics tables from system memory to display memory over the VME bus, while dynamic display memory manipulation is executed by the graphics processor. This external port shared access is very effective for the multi-windowing Desktop Manager type environment.

1.2.1.4 VME Interface

The VME-Graphics Subsystem interfaces with the VME bus as a 24 address bit, 16 or 32 data bit slave. The VME-Graphics Subsystem host interface logic provides interfacing capability consistent with the VME specification for the following VME-defined functional modules:

- Slave — This is the ability to respond to an access attempt by a master. Determination of an attempt to access is based on recognition of a certain address, set of addresses, or address range. The VME-Graphics Subsystem responds to two base addresses (Base Address 1 for the display memory board and Base Address 2 for the display controller board) located on any one-megabyte boundary within an address range of 700000 to E00000.
- Interrupter — The interrupter performs three tasks. It asserts the interrupt request line, supplies a status/ID (vector) byte to the data bus when its request has been acknowledged, and propagates the interrupt acknowledge daisy chain signal if it is not requesting that level of interrupt. There are seven levels (1-7) of interrupt request priority supported by VME, with Level 7 being the highest. The interrupt priority level of the VME-Graphic Subsystem is jumper selectable from VME interrupt Levels 4 through 6, Levels 1, 2, 3, and 7 are not selectable.

For detailed information regarding the VME bus interface and the associated functional units, refer to the *VMEbus Specification Manual*, Motorola part number MVMEBS/D1.

1.2.2 Display Memory

The display memory is a dual-port 512 Kbyte RAM that is organized as 64K x 64 (see Figure 1-3). The video logic picks up a full 64 bits over the refresh port on each video refresh cycle, while the VME bus and graphics processor accesses via the external port consist of either 16 or 32 bits. Byte accesses from the VME bus are not supported.† As previously discussed, arbitration of display memory access is a function of the memory access manager located on the display controller board.

The display memory board provides 512 Kbytes of dual frame buffer memory for monochrome video operations. The memory is structured into two frame buffers of 1280 by 1024 pixels with an additional 192 Kbytes of non-displayable memory available for storing menus, fonts, and display tables. This double buffering concept enables the user to display an image from one frame buffer while updating the second frame buffer.

† Byte accesses do not cause bus errors as defined in the VME Specification.

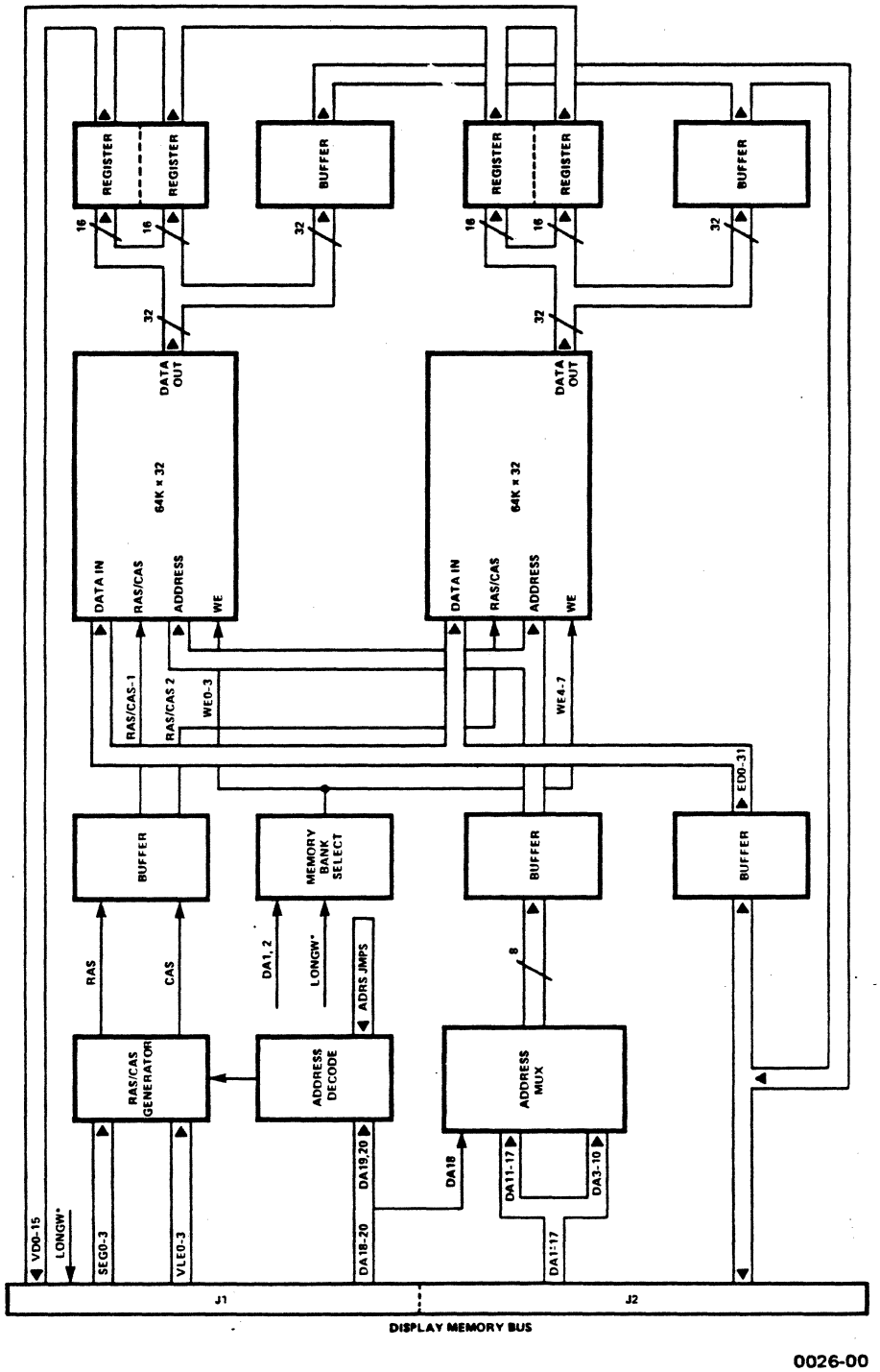

Figure 1-3. Display Memory Block Diagram

SECTION 2: SPECIFICATIONS

This section provides performance specifications and operating requirements for the Monochrome VME-Graphics Subsystem.

2.1 Form Factor

Both the display controller board and the display memory board are standard double-wide VME specification, 160mm x 233.33mm.

2.2 Power Requirements

The VME-Graphics Subsystem power requirements (worst case) are shown in Table 2-1.

Table 2-1. Power Requirements

Board	Voltage	Current
Display Controller	+5 volts	8 amps
Display Controller	-12 volts	0.6 amps
Display Memory	+5 volts	3.5 amps

2.3 Environmental

The environmental requirements for the VME-Graphics Subsystem are as follows:

Temperatures:

0 to 50 degrees centigrade (operating)
-40 to 65 degrees centigrade (non-operating)

Humidity:

10 to 90 percent (Non-condensing)

2.4 System Bus

The VME-Graphics Subsystem interfaces with the VME bus as defined in the *VMEbus Specification Manual*, Motorola part number MVMEBS/D1.

The display controller board is attached to the VME bus via connectors P1 and P2. The pin assignments and signal mnemonics for the display controller board connectors P1 and P2 are provided in Tables 2-2 and 2-3, respectively.

The display memory board has one VME bus connector, P1. The logical connections from the VME bus to the display memory board via connector P1 are limited to the following:

1. GROUND — A9, C9, A11, A15, A17, A19, B20, and B23
2. +5 Volts — A32, B32, and C32
3. SYSRES — C12
4. On Rev. B boards (and later) BGIN and BGOUT as well as IACKIN and IACKOUT are tied together.

All other signals from the VME bus to the display memory board must go through the display controller board.

Table 2-2. Display Controller Connector P1 Pin Assignments

Pin Number	Row A Signal Mnemonic	Row B Signal Mnemonic	Row C Signal Mnemonic
1	VMED0	BBUSY*†	VMED8
2	VMED1	BCLR*†	VMED9
3	VMED2	ACFAIL*†	VMED10
4	VMED3	BG0IN*††	VMED11
5	VMED4	BG0OUT*††	VMED12
6	VMED5	BG1IN*††	VMED13
7	VMED6	BG1OUT*††	VMED14
8	VMED7	BG2IN*††	VMED15
9	GND	BG2OUT*††	GND
10	SYSCLK†	BG3IN*††	SYSFAIL*†
11	GND	BG3OUT*††	VMEBERR*†
12	VMEDS1*	BR0*†	SYSRESET*
13	VMEDS0*	BR1*†	LWRD*
14	VMEWR*	BR2*†	VMEAM5
15	GND	BR3*†	VADD23
16	VMEDTACK*	VMEAM0†	VADD22
17	GND	VMEAM1†	VADD21
18	VMEAS*	VMEAM2	VADD20
19	GND	VMEAM3	VADD19
20	VMEIACK*	GND	VADD18
21	IACKIN*	SERCLK†	VADD17
22	IACKOUT*	SERDAT†	VADD16
23	VMEAM4	GND	VADD15
24	VADD7	IRQ7*†	VADD14
25	VADD6	IRQ6*	VADD13
26	VADD5	IRQ5*	VADD12
27	VADD4	IRQ4*	VADD11
28	VADD3	IRQ3*†	VADD10
29	VADD2	IRQ2*†	VADD9
30	VADD1	IRQ1*†	VADD8
31	-12V	+5V STDBY†	+12V†
32	+5V	+5V	+5V

NOTE

An asterisk following a signal name indicates that the signal is true when the signal is low.

† VME bus signals, but no connection on display controller board.

†† VME BUS GRANT signal lines are connected on Revision D display controller boards. Earlier versions of the board do not have these lines connected.

Table 2-3. Display Controller Connector P2 Pin Assignments

Pin Number	Row A Signal Mnemonic	Row B Signal Mnemonic	Row C Signal Mnemonic
1	n/c†	+5V	n/c
2	n/c	GND	n/c
3	n/c	n/c	n/c
4	n/c	n/c	n/c
5	n/c	n/c	n/c
6	n/c	n/c	n/c
7	n/c	n/c	n/c
8	n/c	n/c	n/c
9	n/c	n/c	n/c
10	n/c	n/c	n/c
11	n/c	n/c	n/c
12	n/c	GND	n/c
13	n/c	+5V	n/c
14	n/c	D16	n/c
15	n/c	D17	n/c
16	n/c	D18	n/c
17	n/c	D19	n/c
18	n/c	D20	n/c
19	n/c	D21	n/c
20	n/c	D22	n/c
21	n/c	D23	n/c
22	n/c	GND	n/c
23	n/c	D24	n/c
24	n/c	D25	n/c
25	n/c	D26	n/c
26	n/c	D27	n/c
27	n/c	D28	n/c
28	n/c	D29	n/c
29	n/c	D30	n/c
30	n/c	D31	n/c
31	n/c	GND	n/c
32	n/c	+5V	n/c

† n/c means not connected on the display controller board

2.5 Display Memory Bus

The communication between the display controller board and display memory board is via the display memory bus. This bus is implemented with two 50-pin ribbon cables (see Figure 2-1). Pin assignments for the associated connectors, J1 and J2, are given in Tables 2-4 and 2-5.

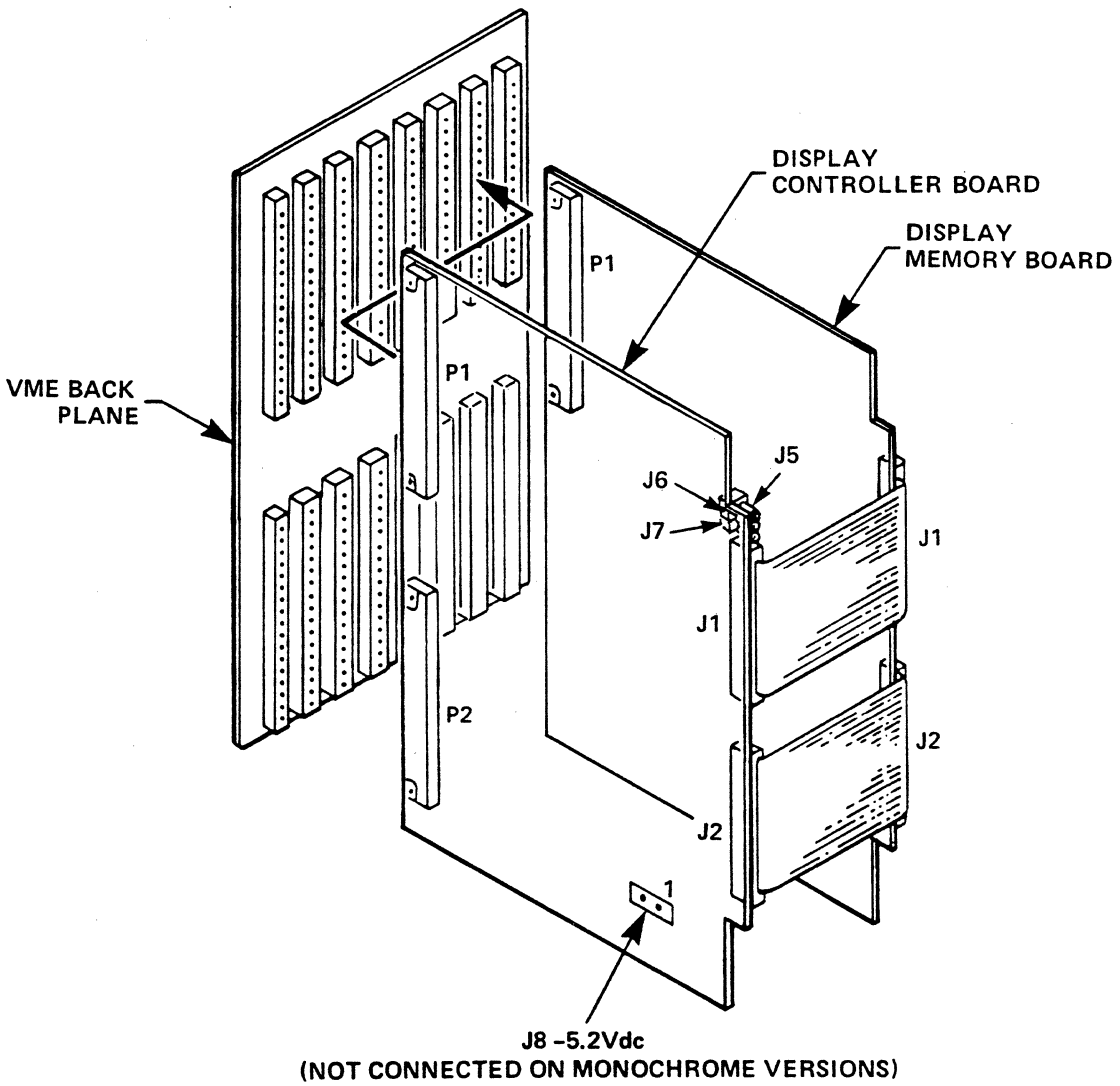

Figure 2-1. VME-Graphics Subsystem Connectors

Table 2-4. Display Memory Bus Connector J1 Pin Assignments

Pin Number	Signal Mnemonic	Signal Name	Pin Number	Signal Mnemonic	Signal Name
1	GND	Ground	26	GND	Ground
2	UCMD*†	Microcommand*	27	VD15	Video Data 15
3	USEWE*†	Use WE Register*	28	VD14	Video Data 14
4	WESEL*†	WE Register Select*	29	GND	Ground
5	ALLPL*†	Write All Planes*	30	VD13	Video Data 13
6	DA19	Display Address 19	31	VD12	Video Data 12
7	DA18	Display Address 18	32	GND	Ground
8	EAC*	External Acknowledge*	33	VD11	Video Data 11
9	DWT*	Data Write*	34	VD10	Video Data 10
10	EAS*	Enable Address Strobe*	35	GND	Ground
11	DSB*	Data Strobe*	36	VD9	Video Data 9
12	TC2†	Timing Clock 2	37	VD8	Video Data 8
13	VLE3*	Video Latch Enable 3*	38	GND	Ground
14	GND	Ground	39	VD7	Video Data 7
15	VLE2*	Video Latch Enable 2*	40	VD6	Video Data 6
16	GND	Ground	41	GND	Ground
17	VLE1*	Video Latch Enable 1*	42	VD5	Video Data 5
18	GND	Ground	43	VD4	Video Data 4
19	VLE0*	Video Latch Enable 0*	44	GND	Ground
20	GND	Ground	45	VD3	Video Data 3
21	SEG3*	Segment 3*	46	VD2	Video Data 2
22	SEG2*	Segment 2*	47	GND	Ground
23	GND	Ground	48	VD1	Video Data 1
24	SEG1*	Segment 1*	49	VD0	Video Data 0
25	SEG0*	Segment 0*	50	LONGW*	Longword*

NOTE

The J1 pin assignments provided in Table 2-4 are for Revision D display controller boards. Earlier versions of the board have slightly different pin assignments for connector J1. These earlier pin assignments are given in Appendix A.

† These signals are ignored by the monochrome display memory board.

Table 2-5. Display Memory Bus Connector J2 Pin Assignments

Pin Number	Signal Mnemonic	Signal Name	Pin Number	Signal Mnemonic	Signal Name
1	GND	Ground	26	ED6	External Data 6
2	ED30	External Data 30	27	ED5	External Data 5
3	ED29	External Data 29	28	ED4	External Data 4
4	ED28	External Data 28	29	ED3	External Data 3
5	ED27	External Data 27	30	ED2	External Data 2
6	ED26	External Data 26	31	ED1	External Data 1
7	ED25	External Data 25	32	ED0	External Data 0
8	ED24	External Data 24	33	DA17	Display Address 17
9	ED23	External Data 23	34	DA16	Display Address 16
10	ED22	External Data 22	35	DA15	Display Address 15
11	ED21	External Data 21	36	DA14	Display Address 14
12	ED20	External Data 20	37	DA13	Display Address 13
13	ED19	External Data 19	38	DA12	Display Address 12
14	ED18	External Data 18	39	DA11	Display Address 11
15	ED17	External Data 17	40	DA10	Display Address 10
16	ED16	External Data 16	41	DA9	Display Address 9
17	ED15	External Data 15	42	DA8	Display Address 8
18	ED14	External Data 14	43	DA7	Display Address 7
19	ED13	External Data 13	44	DA6	Display Address 6
20	ED12	External Data 12	45	DA5	Display Address 5
21	ED11	External Data 11	46	DA4	Display Address 4
22	ED10	External Data 10	47	DA3	Display Address 3
23	ED9	External Data 9	48	DA2	Display Address 2
24	ED8	External Data 8	49	DA1	Display Address 1
25	ED7	External Data 7	50	ED31	External Data 31

The signal functions for the display memory bus are as follows:

- UCMD* — MICROCOMMAND* has no effect in the monochrome graphics subsystem.
- USEWE* — USE WRITE ENABLE REGISTER* has no effect in the monochrome graphics subsystem.
- WESEL* — WRITE ENABLE SELECT has no effect in the monochrome graphics subsystem.
- ALLPL* WRITE ALL PLANES has no effect in the monochrome graphics subsystem.
- TC2 — TIMING CLOCK 2 has no effect in the monochrome graphics subsystem.
- DA1-19 — DISPLAY ADDRESS 1 through 19 are the display address lines. Display addresses may be generated on these lines by the display address generator, VME bus, or the graphics processor.
- EAC* — The EXTERNAL ACKNOWLEDGE* signal is generated by the display memory board to acknowledge access on the external port. In the case of a read operation, the EAC* indicates that there is valid data on the external data bus.
- DWT* — DATA WRITE* is generated by the display controller logic. DWT* develops the display memory WRITE ENABLE signals. When DWT* is asserted, the external display memory access is a Write operation. When DWT* is false, the external access is a Read operation.

- EAS* — EXTERNAL ADDRESS STROBE* is generated by the display controller logic. EAS* in conjunction with the decoded DISPLAY ADDRESS lines develop the display memory on-board ADDRESS STROBE signal.
- DSB* — DATA STROBE* is generated by the display controller logic. A true DSB* signal indicates that there is valid data on the External Data bus.
- VLE0-3 — The VIDEO LATCH ENABLE signals are generated by the display memory access management logic on the display controller board. Each time VLE is asserted, a 16-bit data transfer from a selected display memory segment takes place from display memory to the display refresh shift register on the display controller board.
- SEG0-3 — The SEGMENT signals are generated by the display memory access management logic on the display controller board. Each time SEG is asserted, VIDEO LATCH ENABLE signals can instigate data transfers from display memory to the display refresh shift registers for that segment of memory.
- VD0-15 — VIDEO DATA 0 through 15 are the display refresh data lines.
- LONGW* — LONGWORD* is generated by the display controller logic as a function of the VME bus signal, LWRD*. LONGW* is asserted in order to make a 32-bit (longword) data transfer between display memory and the VME bus via the external data bus.
- ED0-31 — EXTERNAL DATA 0 through 31 are the external data bus lines that provide the means for bi-directional data flow between display memory and the graphics processor or the VME bus.

2.6 Video Interface

The signal interface for black and white video operations is provided by a coaxial cable attached to the display controller board connector, J5 (see Figure 2-1). The video interface timing specifications are given in Table 2-6.

Table 2-6. Video Interface Timing

Description	Specification
Screen Format	1280 x 1024
Screen Refresh	63.0 Hz non-interlaced
Vertical Retrace	682.3 microseconds
Horizontal Scan Time	14.83 miroseconds (67.4 KHz)
Horizontal Retrace	3.845 microseconds
Pixel Cell Time	8.6 nanoseconds
Monitor Interface	ECL video and separate TTL VSYNC and HSYNC

2.7 Interrupt Vector

The interrupt vector number is jumper selectable from the display controller board. The interrupt vector range is 128 to 255 (see Section 3 for jumper configurations).

2.8 Display Memory

Display memory is a shared resource between the graphics processor and the VME bus. The display memory board provides 512 Kbytes of memory. Displayable memory is structured in two 1280 x 1024 pages. See Section 4 for more detailed information regarding the display memory address map.

2.9 Addressing

The VME-Graphics Subsystem base address is relocatable to any address from 700000 to E00000 (hex) in increments of one megabyte. Refer to Section 3 for address mapping details and the associated jumper configurations.

2.10 Indicators

There is one LED (DS1) located on the display controller board. When the LED is on, the graphics processor is executing graphics instructions from the FIFO. When off, the LED indicates that the graphics processor is idle.

SECTION 3: CONFIGURATION

The display controller board has jumpers which allow considerable configuration versatility. The jumper configurations for the display controller board are discussed in Subsection 3.1. The current display memory board does not have any jumpers. However, display memory boards prior to Revision B do have jumpers, the settings of which are discussed in Subsection 3.2.

3.1 Display Controller Board Jumpers

There are two versions of the monochrome display controller board. The reference designators and functional descriptions for the configuration jumpers are identical for both versions of the display controller, however, the physical locations are different. For display controller boards that are Revision C and earlier, refer to Figure 3-1. The Revision D (and later) display controller board layout is shown in Figure 3-2. (You can determine which board you have by checking the revision level etched on the solder side of the board.) The jumper functions and configurations are discussed in the paragraphs that follow.

3.1.1 Address Map (E1-E6)

A set of three jumpers determine the base address for both the display memory and the display controller within the 24-bit VME address space. The display memory base address is relocatable on any address from 700000 to E00000 (hex) in increments of one megabyte, but utilizes only the first half of the allocated address space within the one-megabyte range. The corresponding base address for the display controller is assigned to the command FIFO register.

In addition to the command FIFO, the display controller also has a Reset register used to reset the Graphics Subsystem. The Reset register is located at display controller base address +6. The intervening address locations (between the FIFO and Reset registers) are reserved for future use. The display memory address range and associated display controller addresses for the valid address jumper configurations are as follows:

Table 3-1. Base Address Jumpers

E5-E6	E3-E4	E1-E2	Display Memory (hex)	Display Controller (hex)
Jumper	Jumper	Jumper	700000-77FFFE	FE0000-FE0006
Jumper	Jumper	Open	800000-87FFFE	FE4000-FE4006
Jumper	Open	Jumper	900000-97FFFE	FE8000-FE8006
Jumper	Open	Open	A00000-A7FFFE	FEC000-FEC006
Open	Jumper	Jumper	B00000-B7FFFE	FF0000-FF0006
Open	Jumper	Open	C00000-C7FFFE	FF4000-FF4006
Open	Open	Jumper	D00000-D7FFFE	FF8000-FF8006
Open	Open	Open	E00000-E7FFFE	FFC000-FFC006

In addition to decoding the base address, the VME-Graphics Subsystem also decodes the VME address modifier signals AM0 through AM5. The address modifiers to which the Monochrome VME-Graphics Subsystem responds are determined by the programmable array logic (PAL) located on the display controller board. The default value to which the VME-Graphics Subsystem responds is as follows:

- Standard Addressing — AM5, AM4, and AM3 are high
- Supervisory Data Access — AM2 and AM1 are high, while AM0 is low

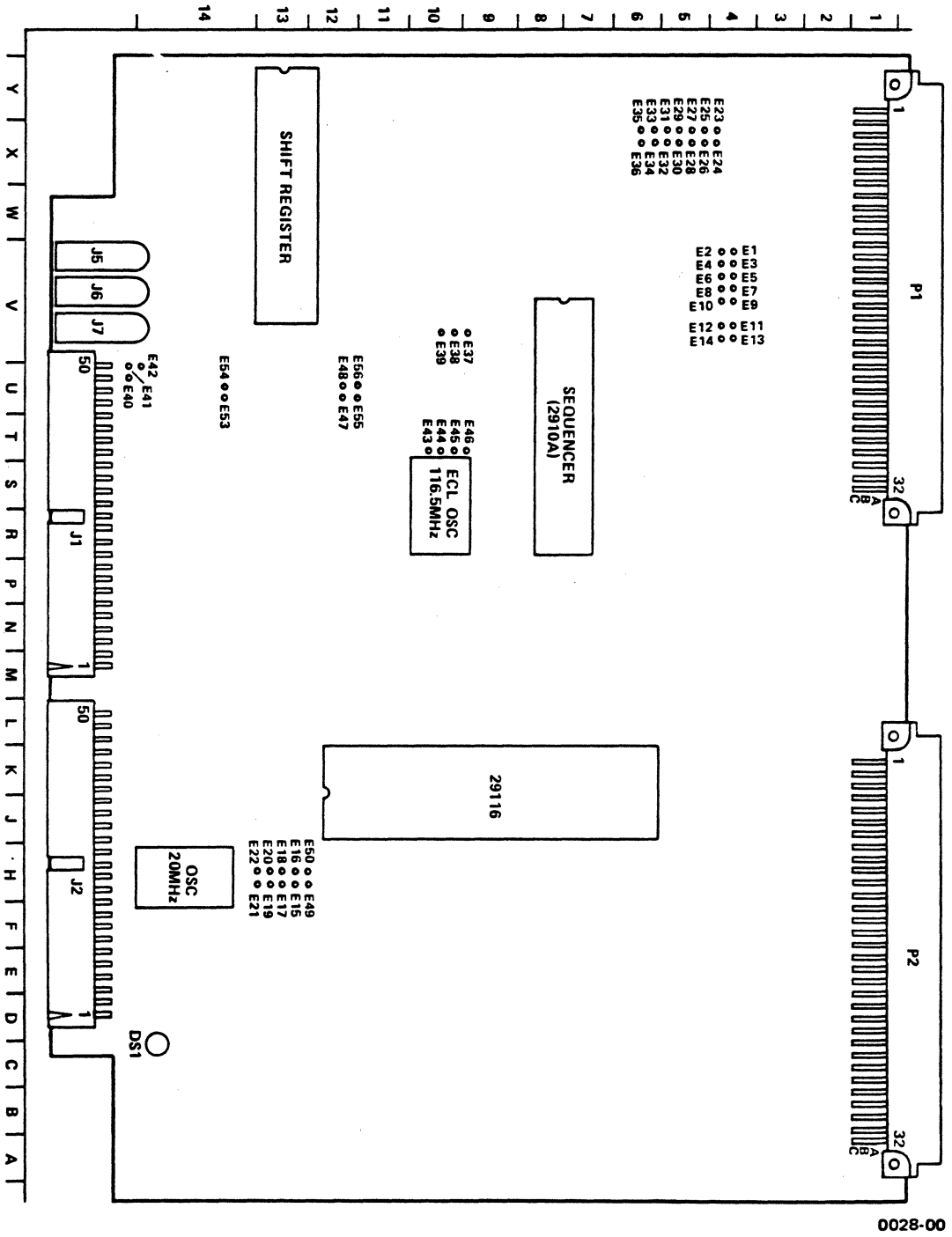

Figure 3-1. Display Controller Board (Rev. C and earlier)

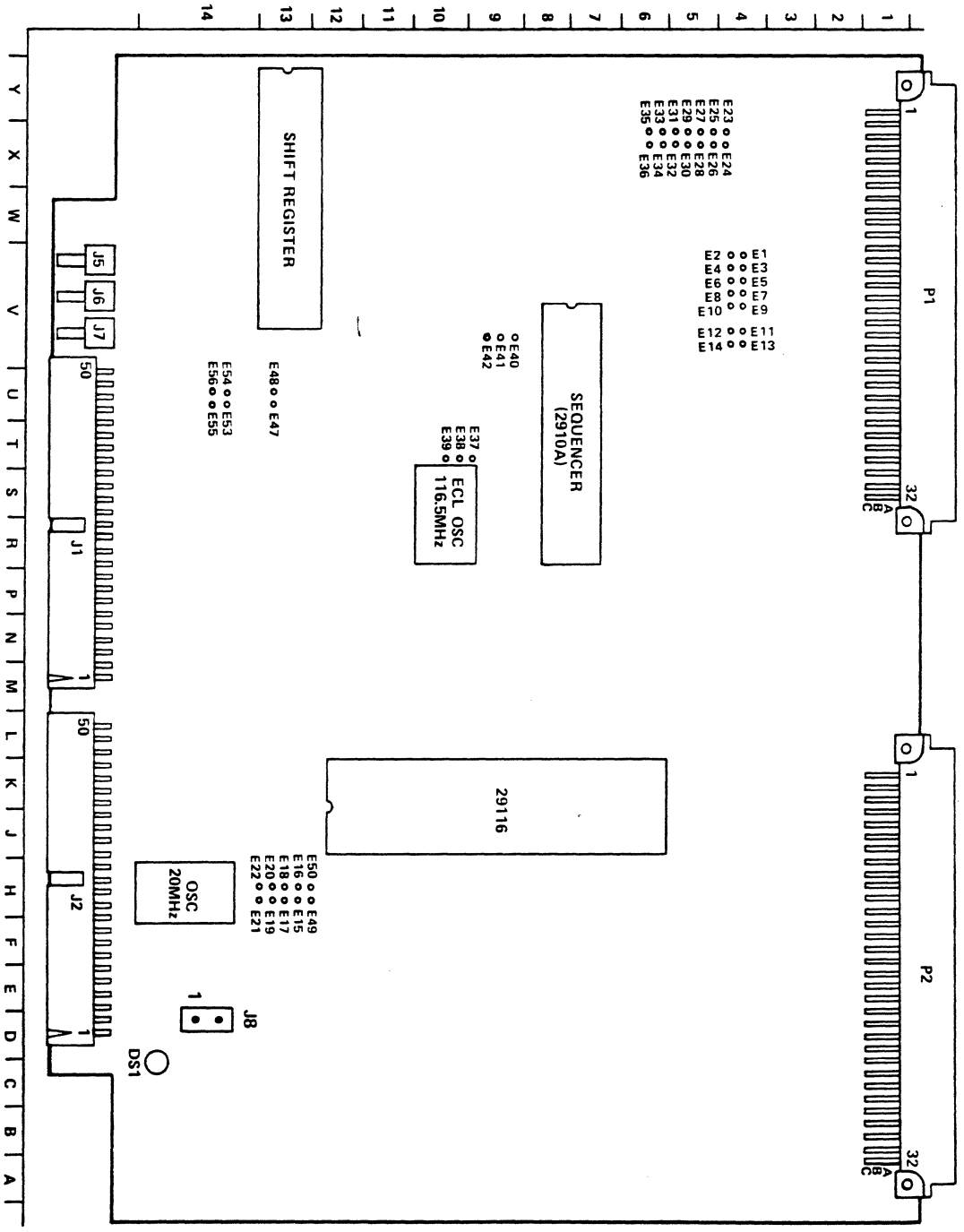

Figure 3-2. Display Controller Board (Rev. D and later)

3.1.2 Interrupt Request Level (E7-E10)

The display controller board is capable of generating interrupt requests at VME priority Levels 4 through 6. Selection of the interrupt request level is the function of jumpers E7 through E10. Table 3-2 gives the jumper configuration for the interrupt request levels.

Table 3-2. Interrupt Request Levels

Interrupt Level	E7-E8	E9-E10
4	Jumper	Open
5	Open	Jumper
6	Open	Open

3.1.3 Sequencer Output Control (E11-E14)

The sequencer output control jumper configuration is factory set from E13 to E14. All other positions must be left open.

3.1.4 Interrupt Vector (E23-E36)

Jumper posts E23 through E36 are grouped as seven jumper pairs that correspond to the lower seven bits in an interrupt vector number byte. The most significant bit of the vector byte is always set to 1. The remaining seven lower bits may be set to 1 (no jumper) or 0 (jumper installed), resulting in an interrupt vector range of 128 to 255. Figure 3-3 shows the jumper pairs as they correspond to their respective interrupt vector bits. (As an example, the jumper configuration in the figure is set to 162 decimal.)

Bit	Jumpers	Value
0	E34○----○E33	0
1	E30○ ○E29	1
2	E28○----○E27	0
3	E24○----○E23	0
4	E36○----○E35	0
5	E32○ ○E31	1
6	E26○----○E25	0
7	No Jumper	1 (always 1)

Figure 3-3. Interrupt Vector Jumpers

3.1.5 ALU Clock Generation (E19-E22, E49, and E50)

The ALU clock generation control jumpers are factory set. Jumpers are set from E21 to E22, as well as from E49 to E50. All other positions must be left open.

3.1.6 Jumpers (E15-E18)

Jumpers E15-E18 functionality is dependent on the display controller board's revision level. For display controller board Rev. C or earlier, these jumpers are associated with software configuration. On Rev. D (or later) display controller boards, the jumpers are associated with monochrome or color video capabilities.

3.1.6.1 Software Configuration (Rev. C and earlier)

The software configuration jumpers are factory set so all positions are open.

3.1.6.2 Color or Monochrome (Rev. D and later)

Jumpers E15-E18 may be configured for either color or monochrome capability. The jumper configuration must match the type of display controller (color or monochrome) as shown in Table 3-3.

Table 3-3. Color or Monochrome Jumper Configurations

Controller	E15-E16	E17-E18
Color	open	jumper
Monochrome	jumper	open

3.1.7 Video Output (E37-E42, E47, and E48)

The video output jumpers are factory set. Jumpers are installed between E37 and E38 as well as between E47 and E48. All other positions are left open.

3.1.8 Jumpers E43-E46

Jumpers E43-E46 are only present on display controller boards prior to Revision D. These jumpers are factory set and must remain open.

3.1.9 Vertical and Horizontal SYNC (E53-E56)

These jumpers are factory set with E53 jumpered to E54 and E55 jumpered to E56.

3.2 Display Memory Board Jumpers

Monochrome display memory boards prior to Revision B have 24 jumper posts (see Figure 3-4). The configurations are defined in the paragraphs that follow.

NOTE

Current monochrome display memory boards (Revision B) have no jumpers.

3.2.1 Memory Plane Select (E1-E8)

The memory plane select jumpers are factory set. A jumper is installed from E1 to E2. All other positions are open.

3.2.2 Video Latch Enable (E9-E16)

The video latch enable jumpers are factory set. A jumper is set from E15 to E16. All other positions are open.

3.2.3 Address Strobe (E17-E24)

The address strobe jumpers are factory set. A jumper is set from E23 to E24. All other positions are open.

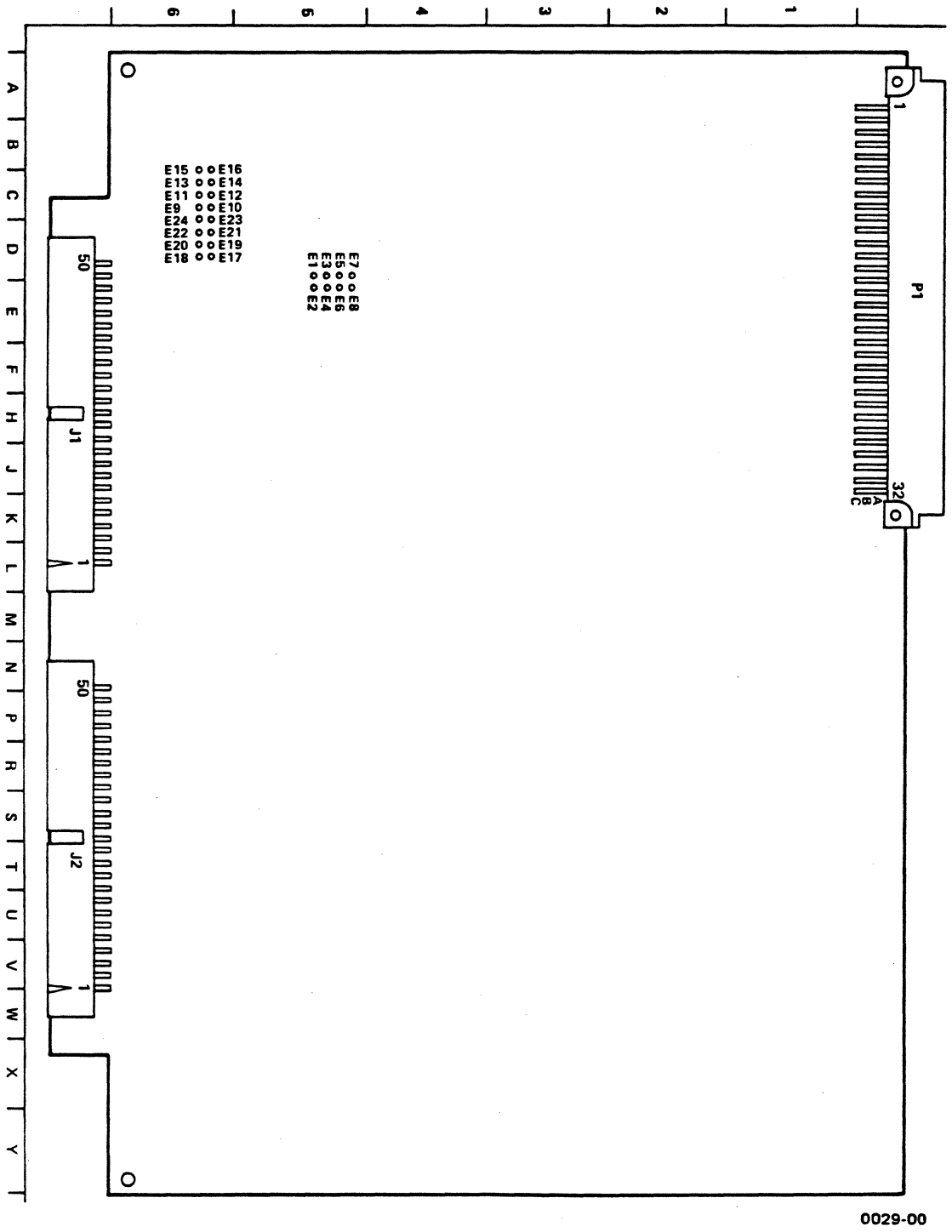

Figure 3-4. Display Memory Board (Prior to Rev. B)

SECTION 4: SOFTWARE INTERFACE

This section provides information regarding the software interface between the VME-Graphics Subsystem and host CPU. Specifically, display memory, control registers, and the microcode instruction set are discussed in the subsections that follow.

4.1 Display Memory

For the black and white VME-Graphics Subsystem, the single display memory plane (Plane 0) is located at Base Address 1. As defined in Section 3.1.1, the Base Address 1 is relocatable on one-megabyte boundaries within the address range of 700000 to E00000 (see Section 3.1.1 for the appropriate jumper selection).

From the point of view of the host processor, the display memory appears as a linear set of addresses ranging from the base address to base address + 7FFFE hex for a single memory plane. However, the display memory appears as a two-dimensional image on the screen, not all of which is visible. Table 4-1 lists the addresses of the display memory Plane 0 with respect to the VME address space and the graphics processor address space. They are different because the graphics processor deals only with 32-bit words.

Table 4-1. Display Memory Address Map

Display Memory Region	VME Start Address (hex)	Graphics Processor Start Address(hex)	Memory Size
Page 0	Base Address 1	0	160 Kbytes
Undisplayed Area 0	Base Address 1 + 28000	A000	96 Kbytes
Page 1	Base Address 1 + 40000	10000	160 Kbytes
Undisplayed Area 1	Base Address 1 + 68000	1A000	96 Kbytes
Total			512 Kbytes

Mapping from linear address space to the two-dimensional screen address space requires the user to know the pitch. Pitch is defined as the offset between adjacent vertical words on the screen. Since the display resolution is fixed at 1280 horizontal pixels, this means that the pitch is 80 if the addressing increment is in words (1280 ÷ 16 = 80). For example, with the upper left corner of the screen being Base Address 1 + 0, the VME address of the word vertically beneath that word is Base Address 1 + A0 hexadecimal (80 decimal words).†

4.2 Control Registers

The two VME-Graphics Subsystem control registers are the FIFO register and the Reset register.

† This example expresses pitch in terms of VME bus addressing. The graphics processor addresses in longwords (32 bits). Consequently, for the same example from the graphics processor view point, the address of the word (longword) vertically beneath Base Address 1 + 0 is 40 decimal.

4.2.1 FIFO Register

The FIFO register is located at Base Address 2. The location of Base Address 2 is dependent on that of Base Address 1 (see Table 3-1 in Section 3.1.1). The FIFO register is used by the host to write graphics commands and parameters to the graphics processor. Unlike the display memory, the FIFO may be accessed at any time. The FIFO has a capacity of 64 words where a word is 16 bits. Byte and longword transfers are not supported, but do not cause bus errors per the VME specification. In Rev. D boards (and later), the FIFO also responds to Base Address 2 + 8 hexadecimal. This is provided so that software can distinguish a Rev. D board from its predecessors.

The user may obtain the status of the FIFO by performing a Read operation at this location. Bit 7 signifies that the FIFO is empty. If Bit 7 is 1, the FIFO is completely empty and is ready to receive more commands and parameters. Bit 6 signifies that the FIFO has at least one empty space into which the user may write. If Bit 6 is set to 1, the FIFO is completely full and cannot accept any more parameters. Should a Write operation be performed while the FIFO is full, data would be lost. Bit 5 shows the status of the graphics processor. If it is 1, the processor is busy and the VME bus cannot be accessed.

NOTE

Bits 5 and 6 are available only with Rev. D boards (and later).

4.2.2 Reset Register

The display controller board has a special address location called RESET (Base Address 2 + 0006 hex). When any data is written to this location, an internal reset of the microprocessor is performed. This has the same effect as issuing an INIT microcode instruction as described in Section 4.3.

4.3 Microcode Instruction Set

The microcode instruction set has been designed to be optimally used by the window manager to operate with the monochrome display monitor. The display memory contains two displayable pages with 1280 x 1024 pixel resolution.

All address pointers (such as *mask*, *src*, or *dest*) describe 32-bit (longword) locations in the display memory. Because 16 bits (the size of the input parameters) are not enough to address all the memory that is available, the bits are split into higher and lower address parameters labled *xxx_h* and *xxx_l*. Only the lower 3 bits of the higher address parameter have any meaning. Bit 0 determines which page of the memory plane is addressed, while bits 1 and 2 determine which plane is addressed.

The parameters are checked for validity when the command is processed by the micrcode. Parameters x, y, w and h must be between 0 and the respective maximum screen size. Only the upper (*xxx_h*) address pointers are checked to ensure that the upper 13 bits are zero. Any invalid commands are ignored. If any of the command parameters are invalid, that command is ignored.

The microcode instruction set includes the following:

alu_clear, alu_set, alu_toggle
The image processor is configured to perform logical operations when filling is taking place. Once one of these commands is issued, the mode selected remains in effect until another ALU mode command is issued. The modes supported are Clear, Set, and Toggle.

box (dest_h, dest_l, x, y, w, h)
Draws a filled box of width "w" and height "h" with the upper left corner at (x, y). The "dest" pointer defines where in the display memory the origin point (0,0) is. Dest_h and dest_l set to 0 will define the origin point to the upper left hand corner of page 0. The "dest" pointer addresses 32-bit words. ALU modes of Clear, Set, and Toggle result in black, white, and complemented boxes, respectively.

boxc (src_h, src_l, xs, ys, dest_h, dest_l, xd, yd, w, h)
Copies a rectangular region of width "w" and height "h" with upper left corner at (xs, ys) to another region of the same size with upper left corner (xd, yd). The "src" and "dest" parameters have the same function as described in the box instruction. ALU mode has no effect.

boxp (pattern_h, pattern_l, xphase, yphase, dest_h, dest_l, x, y, w, h)
Draws a 32 x 32 pattern located at "pattern" into a rectangular region of width "w" and height "h" with upper left corner at (x, y). "Xphase" and "yphase" specify a screen coordinate that would contain the upper left corner of the pattern if it were extended to cover that point. ALU mode has no effect.

clip (x, y, w, h)
Set clipping window region of width "w" and height "h" with upper left corner at (x, y). When clipping is on, no pixels outside this region will be drawn.

clip_on clip_off
Clipping is turned off and on with these commands. When clipping is turned off, no check is made for off screen access. This is useful in situations where it is known that this cannot occur and maximum performance is desired.

init
The screen is cleared, page 0 is selected, ALU mode is set to clear, and clipping window is set to the display memory size with clipping turned off. Interrupts are disabled.

intrp_on, intrp_off
These commands determine whether or not an interrupt is issued to the host when the FIFO is empty and the graphics processor is waiting for more commands. This is the only reliable way for the host to determine that the FIFO is empty and is safe to access the display memory. If the interrupts were previously turned off and then turned on, an interrupt will be issued immediately if the FIFO is empty. When the board is first initialized, interrupts are disabled.

paint (mask_h, mask_l, mx, my, dest_h, dest_l, rx, ry, w, h)
Paints a rectangular region of width "w" and height "h" with upper left corner at (rx, ry). The rectangle is set, cleared, or complemented depending upon the current ALU mode where a bit is set in the mask rectangle. The mask rectangle is assumed to be the same size as the destination rectangle and is located at (mx, my) which is relative to the "mask_hl" pointer.

pgsel (i)
Selects which page (0 or 1) is to be displayed. The selected page will be displayed starting with the subsequent video refresh cycle.

src_pitch (width), dst_pitch (width)
Sets the size of the pitch to be "width" words. A word is defined as 32 bits. The default width is 40 words, which is the width of the screen (40 x 32 = 1280). The minimum and maximum width allowed is 1 and 64, respectively. If a bad width is received, the command is aborted and the previous pitch will still be in effect. Pitch is used to map between linear addresses of the display memory and the two dimensional (x,y) addressing used by the graphics software. The width of the pitch determines the offset between two adjacent words in the "y" direction. Src_pitch affects any command that uses source addresses. This includes "src", "pattern", "mask" and "font" address pointers. Similarly, dst_pitch affects the "dest" address pointer.

vector (dest_h, dest_l, x0, y0, x1, y1)
Draws a vector from (x0, y0) to (x1, y1). ALU mode determines whether black, white, or complemented vectors are drawn. Microcode Instruction OpCode (Table 4-2) shows the codes assigned to each of the microcode instructions.

Table 4-2. Microcode Instruction OpCodes

Instruction	OpCode (hex)
alu_clear	8001
alu_set	8002
alu_toggle	8003
box	8004
boxc	8005
boxp	8006
clip	8007
clip_on	8008
clip_off	8009
init	800A
intrp_on	800B
intrp_off	800C
paint	800D
pgsel	800E
src_pitch	8010
dst_pitch	8011
vector	8013

APPENDIX A: DISPLAY CONTROLLER J1 PIN ASSIGNMENTS

This appendix contains the J1 connector pin assignments for the display controller boards prior to Revision D (see Table A-1).

Table A-1. Connector J1 Pin Assignments

Pin Number	Signal Mnemonic	Signal Name	Pin Number	Signal Mnemonic	Signal Name
1	GND	Ground	26	GND	Ground
2	—	Not used	27	VD15	Video Data 15
3	—	Not used	28	VD14	Video Data 14
4	—	Not used	29	GND	Ground
5	—	Not used	30	VD13	Video Data 13
6	DA19	Display Address 19	31	VD12	Video Data 12
7	DA18	Display Address 18	32	GND	Ground
8	EAC*	External Acknowledge*	33	VD11	Video Data 11
9	DWT*	Data Write*	34	VD10	Video Data 10
10	EAS*	Enable Address Strobe*	35	GND	Ground
11	DSB*	Data Strobe*	36	VD9	Video Data 9
12	GND	Ground	37	VD8	Video Data 8
13	VLE3*	Video Latch Enable 3*	38	GND	Ground
14	GND	Ground	39	VD7	Video Data 7
15	VLE2*	Video Latch Enable 2*	40	VD6	Video Data 6
16	GND	Ground	41	GND	Ground
17	VLE1*	Video Latch Enable 1*	42	VD5	Video Data 5
18	GND	Ground	43	VD4	Video Data 4
19	VLE0*	Video Latch Enable 0*	44	GND	Ground
20	GND	Ground	45	VD3	Video Data 3
21	SEG3*	Segment 3*	46	VD2	Video Data 2
22	SEG2*	Segment 2*	47	GND	Ground
23	GND	Ground	48	VD1	Video Data 1
24	SEG1*	Segment 1*	49	VD0	Video Data 0
25	SEG0*	Segment 0*	50	LONGW*	Longword*

The signal functions for the display memory bus are as follows:

- DA1-19 — DISPLAY ADDRESS 1 through 19 are the display address lines. Display addresses may be generated on these lines by the display address generator, VME bus, or the graphics processor.
- EAC* — The EXTERNAL ACKNOWLEDGE* signal is generated by the display memory board to acknowledge access on the external port. In the case of a read operation, the EAC* indicates that there is valid data on the external data bus.

- DWT* — DATA WRITE* is generated by the display controller logic. DWT* develops the display memory WRITE ENABLE signals. When DWT* is asserted, the external display memory access is a Write operation. When DWT* is false, the external access is a Read operation.
- EAS* — EXTERNAL ADDRESS STROBE* is generated by the display controller logic. EAS* in conjunction with some decoded DISPLAY ADDRESS lines develop the display memory on-board ADDRESS STROBE signal.
- DSB* — DATA STROBE* is generated by the display controller logic. DSB* controls logic on the display memory board, resulting in DTACK* termination.
- VLE0-3 — The VIDEO LATCH ENABLE signals are generated by the display memory access management logic on the display controller board. Each time VLE is asserted, a 16-bit data transfer from a selected display memory segment takes place from display memory to the display refresh shift register on the display controller board.
- SEG0-3 — The SEGMENT signals are generated by the display memory access management logic on the display controller board. Each time SEG is asserted, VIDEO LATCH ENABLE signals can instigate data transfers from display memory to the display refresh shift registers for that segment of memory.
- VD0-15 — VIDEO DATA 0 through 15 are the display refresh data lines.
- LONGW* — LONGWORD* is generated by the display controller logic as a function of the VME bus signal, LWRD*. LONGW* is asserted in order to make a 32-bit (longword) data transfer between display memory and the VME bus via the external data bus.
- ED0-31 — EXTERNAL DATA 0 through 31 are the external data bus lines that provide the means for bi-directional data flow between display memory and the graphics processor or the VME bus.