VAXstation 2000 and MicroVAX 2000 Technical Manual

Order Number EK-VTTAA-TM-001

digital equipment corporation
maynard, massachusetts

First Edition, July 1987

Contents

About This Manual

Chapter 1 System Introduction

Chapter 2 Functional System Overview

Chapter 3 VS410 System Module Detailed Description

Chapter 4 MS400 Option Memory Modules

Chapter 5 ThinWire Ethernet (DESVA) Option Module

Chapter 6 Resistor Load Module

Chapter 7 Power Supply

Chapter 8 Drives

Chapter 9 DEC423 Converter (MicroVAX 2000)

Chapter 10 Expansion Peripherals

Appendix A Timing Diagrams

Appendix B Physical Address Maps

Figures

Tables

About This Manual

This manual documents system design concepts and hardware functions for the VAXstation 2000 and MicroVAX 2000 computer systems. It describes options that support the systems, and provides hardware programming information.

Refer to the Reference Manuals section for a listing of documents that apply to the VAXstation 2000 and MicroVAX 2000 computer systems.

ORGANIZATION

The manual is divided into ten chapters and two appendices.

Chapter 1 - System Introduction describes the VAXstation 2000 and MicroVAX 2000 systems. It also lists the physical characteristics of the components that make up both systems.

Chapter 2 - Functional System Overview provides a functional overview of the system module in the VAXstation 2000 and the MicroVAX 2000 systems.

Chapter 3 - VS410 System Module Detailed Description explains the system module in detail.

Chapter 4 - MS400 Option Memory Modules describes the MS400-AA and MS400-BA memory modules that are options to the KA410-AA system module.

Chapter 5 - ThinWire Ethernet (DESVA) Option Module describes the option that enables a VAXstation 2000 or MicroVAX 2000 system to connect to an Ethernet network.

Chapter 6 - Resistor Load Module explains the module that is used to regulate the power supply of expansion boxes when less than two drives are installed.

Chapter 7 - Power Supply lists the operating specifications of the H7848 power supply.

Chapter 8 - Drives provides an overview of the drives available for use with VAXstation 2000 and MicroVAX 2000 systems.

Chapter 9 - DEC423 Converter (MicroVAX 2000) describes the physical characteristics of the converter, which permits easy installation of terminals and printers using MMJ connectors.

Chapter 10 - Expansion Peripherals describes the three expansion peripherals available with the VAXstation 2000 and MicroVAX 2000 systems.

Appendix A - System Timing Diagrams displays timing diagrams for the system.

Appendix B - Physical Address Maps lists system module and option module addresses.

REFERENCE MANUALS

Manual	Order Number
VAXstation 2000 Hardware Installation Guide	EK-VAXAA-IN
VAXstation 2000 Owner's Manual	EK-VAXAA-OM
VAXstation 2000/MicroVAX 2000 Maintenance Guide	EK-VSTAA-MG
MicroVAX 2000 Hardware Installation Guide	EK-MVXAA-IN
MicroVAX 2000 Owner's Manual	EK-MVXAA-OM
VR290 Service Guide	ED-VR290-SM
VAXstation 2000, MicroVAX 2000, VAXmate Network Guide	EK-NETAA-UG

NOTES, CAUTIONS, and WARNINGS

Notes, cautions, and warnings appear throughout this book.

- Notes contain general information about a topic.
- Cautions contain information to prevent damage to equipment.
- Warnings contain information to prevent personal injury.

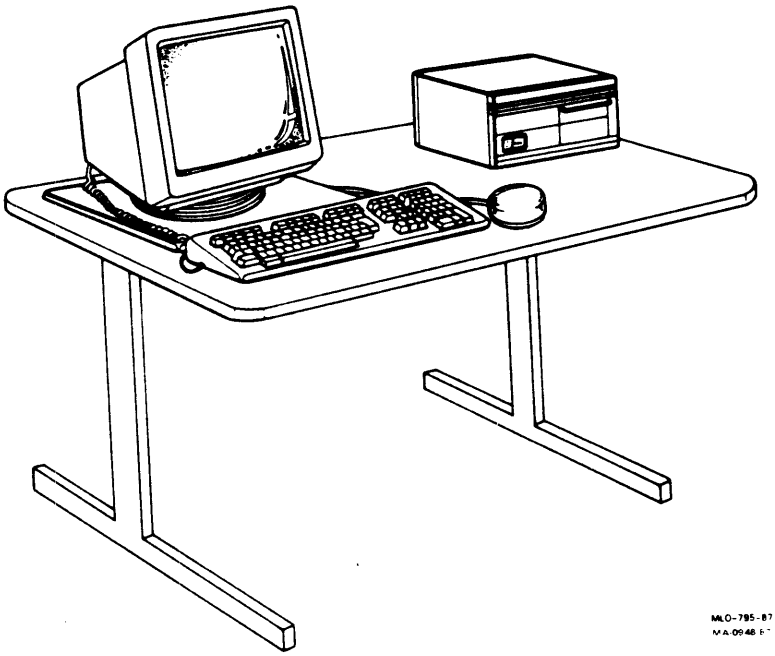
Chapter 1
System Introduction

1.1 VAXstation 2000 System Description

The following paragraphs provide a physical description of the VAXstation 2000 system. The VAXstation 2000 consists of the following four hardware components (Figure 1-1).

- System box
- Video monitor
- Keyboard
- Mouse/Tablet

Figure 1-1: The VAXstation 2000 Computer System



MLO-795-87

1.1.1 VS410 System Box

The VS410 system box contains the following components.

- KA410 system module - The KA410 system module is central to the entire computer system. It is a printed circuit board mounted on the FCC shield. The system module contains all the control and interface electronics needed to support the CPU chip, support all I/O for the disks and tapes, support the video subsystem, and support the three option ports (memory, Ethernet network, and a graphics option port). This system module contains 2 megabytes of RAM and is used in both the VAXstation 2000 and MicroVAX 2000 systems. A jumper setting on the system module determines which system it is configured for.
- MS400 memory option module - The memory module provides two to four additional megabytes of RAM memory. It is a printed circuit board mounted on standoffs on the system module and electrically connected to the system module through two 40-pin connectors. Although the memory module is called an option, additional memory is necessary to run the VMS or ULTRIX operating systems.
- Ethernet network option module - The Ethernet network module provides an IEEE 802.3 interface to the ThinWire Ethernet communications network. It is a printed circuit board mounted on standoffs on the system module and electrically connected to the system module through two 40-pin connectors. This Ethernet network module is an option on the MicroVAX 2000 system but comes standard in the VAXstation 2000 system.
- RX33 floppy diskette drive - The system box may contain an RX33 half-height floppy diskette drive. The RX33 media stores up to 1.23 megabytes of data. This drive is available on both the VAXstation 2000 and MicroVAX 2000 systems.
- RD32 hard disk drive - The system box may contain an RD32 half-height hard disk drive. The RD32 stores up to 40 megabytes of data. This drive is available on both the VAXstation 2000 and MicroVAX 2000 systems.

1.1.2 Video Monitor

The video monitor provides the system display on the VAXstation 2000 system. It is a VR260 monochrome monitor that provides black and white display for the VAXstation 2000 system. The monitor has two display controls on the side panel to adjust brightness and contrast.

1.1.3 LK201 Keyboard

The operator uses the keyboard to enter data into the system. The keyboard contains three keypads (main, editing, and numeric) and a series of special function keys.

1.1.4 VSXXX Mouse

The operator uses the mouse to position the cursor on the monitor screen. The mouse contains three keys and a position movement transducer for positioning the cursor on the display.

1.2 MicroVAX 2000 System Description

This section provides a physical description of the MicroVAX 2000 system. The MicroVAX 2000 consists of the following three hardware components (Figure 1-2).

- System box
- Video console terminal
- Keyboard

1.2.1 VS410 System Box

The VS410 system box contains the same components as listed in Section 1.1.1, plus one additional component. The MicroVAX 2000 system has a DEC423 converter attached to the back of the system box and is mounted over the video and printer ports. The DEC423 converter changes the RS232 signals on the 15-pin video port and 9-pin printer port into DEC423 signals which go out to the three MMJ connectors.

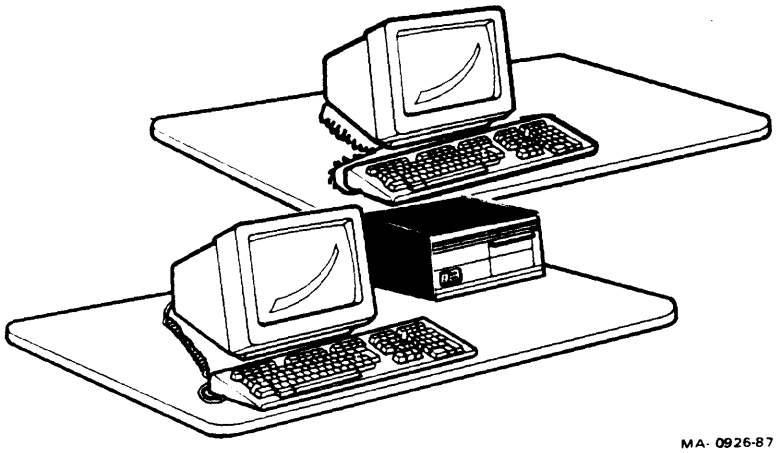
1.2.2 Video Console Terminal

The video console terminal provides the system display. The console terminal is a VT220 which provides a black and white display. It has two display controls for adjusting the brightness and contrast and also has a tilt control on the side panel for adjusting the viewing level.

1.2.3 LK201 Keyboard

The operator uses the keyboard to enter data into the system. The keyboard contains three keypads (main, editing, and numeric) and a series of special function keys.

Figure 1-2: The MicroVAX 2000 Computer System



MA-0926-87

1.3 Physical Characteristics

This section lists the physical characteristics of the components that comprise the VAXstation 2000 and MicroVAX 2000 systems.

1.3.1 System Box

The VS410 system box is housed in a desk top enclosure. All cable access to it is from the rear panel. Cooling air intake is through the front panel and exhaust is through the rear panel. No clearance is required at the top or bottom, or either side of the box.

Width	12.75 inches	323.85 mm
Depth	11.25 inches	285.75 mm
Height	5.5 inches	139.7 mm
Weight	28 pounds	12.7 kg

The dimensions of the VS410 system box with BA40A expansion adapter are as follows.

Width	12.75 inches	323.85 mm
Depth	11.25 inches	285.75 mm
Height	7 inches	177.8 mm
Weight	30 pounds	13.6 kg

1.3.1.1 KA410 System Module

Width	10 inches	254 mm
Length	14 inches	355.6 mm
Height	1.25 inches	32 mm

1.3.1.2 Network Interconnect Module

Width	4 inches	102 mm
Length	7 inches	178 mm
Height	0.25 inches	6.35 mm

1.3.1.3 MS400 Memory Module

Width	4.6 inches	116.84 mm
Length	8 inches	203.2 mm
Height	0.38 inches	9.65 mm

1.3.1.4 Power Supply

Width	4.75 inches	120.65 mm
Length	10.25 inches	260.35 mm
Height	3.75 inches	95.25 mm

1.3.1.5 RX33 Diskette Drive

Width	5.75 inches	146.05 mm
Length	8 inches	203.2 mm
Height	1.69 inches	42.93 mm
Weight	2.9 pounds	1.32 kg

1.3.1.6 RD32 Disk Drive

Width	5.75 inches	146.05 mm
Length	8 inches	203.2 mm
Height	1.63 inches	41.4 mm
Weight	3.5 pounds	1.59 kg

1.3.1.7 DEC423 Converter (MicroVAX 2000)

Width	3 inches	76.2 mm
Length	3.3 inches	83.82 mm
Height	1.23 inches	31.24 mm
Weight	5.6 ounces	159 g

1.3.1.8 Resistor Load Module

Width	4 inches	101.6 mm
Length	7 inches	177.8 mm
Height	0.5 inches	12.7 mm

1.3.2 BA40B Expansion Boxes

The power supply and resistor load modules in the expansion boxes are the same as in the system box. Dimensions of the BA40B storage expansion boxes are as follows.

Width	12.75 inches	323.85 mm
Depth	11.25 inches	285.75 mm
Height	5.5 inches	139.7 mm
Weight	20 pounds	- 9.1 kg

1.3.2.1 RD53 Disk Drive

Width	5.75 inches	146.05 mm
Length	8.2 inches	208.28 mm
Height	3.37 inches	85.6 mm
Weight	6.3 pounds	2.8 kg

1.3.2.2 TZK50 Controller Board

Width	5.7 inches	144.78 mm
Length	8 inches	203.2 mm
Height	0.625 inches	15.88 mm

1.3.2.3 TK50 Tape Drive

Width	5.75 inches	146.05 mm
Length	8.4 inches	213.36 mm
Height	3.25 inches	82.55 mm
Weight	5 pounds	2.27 kg

1.3.3 BA40A Expansion Adapter

Width	12.75 inches	323.85 mm
Length	11.25 inches	285.75 mm
Height	1.5 inches	38.1 mm
Weight	2 pounds	0.9 kg

1.3.3.1 Disk Interface Module

Width	3.2 inches	81.28 mm
Length	5.2 inches	132.08 mm
Height	0.4 inches	10.16 mm

Chapter 2
Functional System Overview

This chapter describes the functional overview of the system module in the VAXstation 2000 and MicroVAX 2000 systems. Functional overviews of the optional modules to these systems are described within their chapter and are not discussed here. Figure 2-1 shows the functional block diagram of the system module.

2.1 Central Processor Overview

The central processor consists of a DC333 MicroVAX CPU chip and a DC337 MicroVAX FPU chip. The DC333 MicroVAX CPU chip is a 32-bit virtual memory microprocessor that implements a subset VAX-compatible central processor. The DC337 FPU chip implements a subset VAX-compatible floating point unit. The FPU chip provides floating point computation capabilities to the MicroVAX CPU chip. Each chip is contained in a 68-pin package and both chips reside on the VS410 system module.

Both chips use the 40 MHz oscillator and communicate to each other over the 32-bit VDAL CPU bus. The F373 latch and the F245 bidirectional bus transceiver buffer the VDAL CPU bus to the ELAD bus and BDAL bus, respectively. Figure 2-2 shows the functional block diagram of the CPU chip and the FPU chip.

Figure 2–1: VS410 System Module Functional Block Diagram

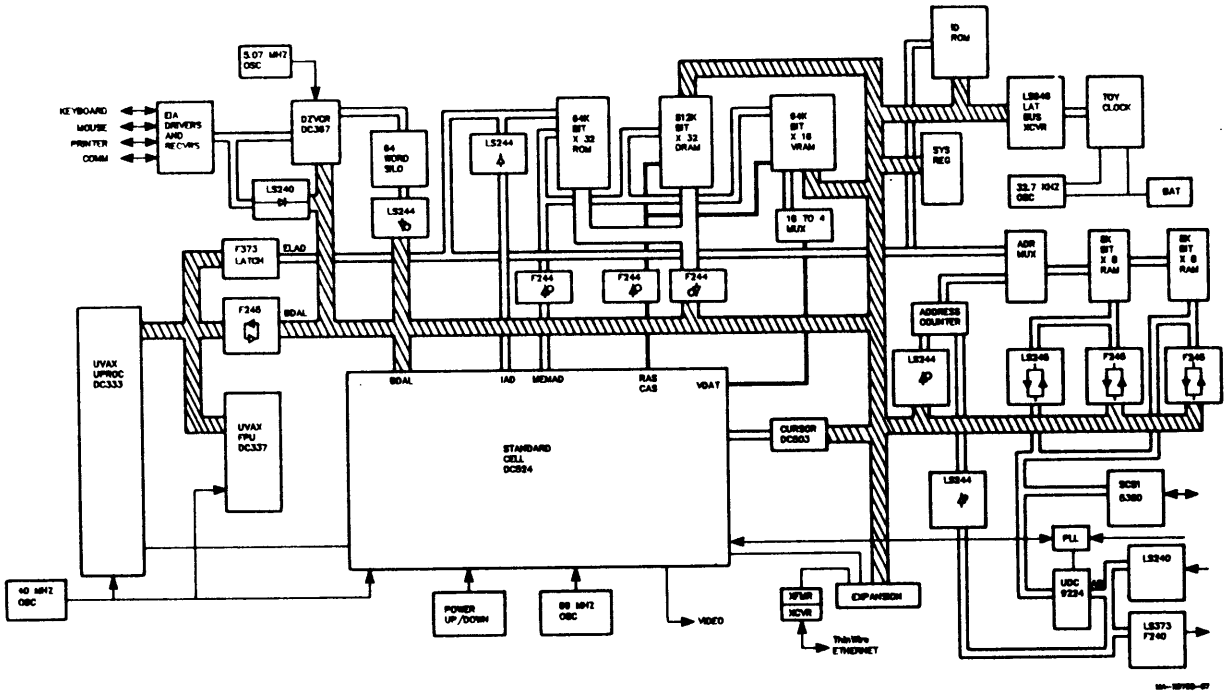
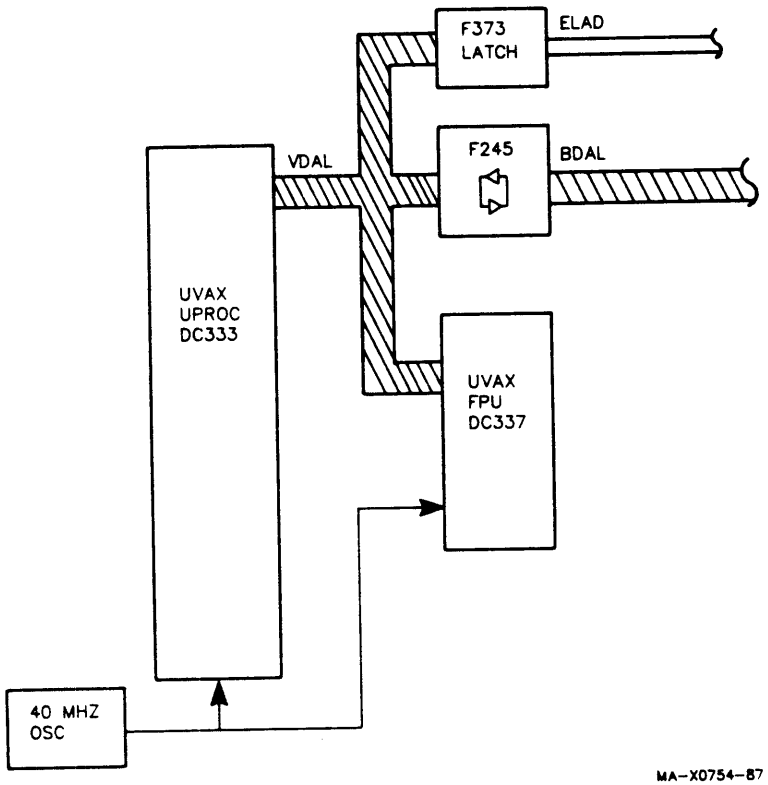


Figure 2-2: Block Diagram of the CPU Chip and the FPU Chip



Key features supported by the DC333 MicroVAX CPU chip:

- Subset VAX data types — The chip supports the following subset of the VAX data types: byte, word, longword, quadword, character string, and variable length bit field. Support for f_floating, d_floating, and g_floating is available via the floating point unit chip. Support for the remaining VAX data types can be provided by macrocode emulation.
- Subset VAX instruction set — The chip implements the following subset of the VAX instruction set: integer and logical, address, variable length bit field, control, procedure call, miscellaneous, queue, MOVC3/MOVC5, and operating system support. Floating point is implemented through the floating point unit chip. The remaining VAX instructions can be implemented via macrocode emulation (the chip provides microcode assists for the emulation of the character string, decimal string, EDITPC, and CRC instructions).

- Floating point — The chip supports f_floating, d_floating, and g_floating data types through the FPU; does not support h_floating.
- Full VAX memory management — The chip includes a demand paged memory management unit which is fully compatible with VAX memory management. System space addresses are virtually mapped through single-level page tables and process space addresses through double-level page tables.
- External interface based on industry standards — The chip's external interface is a 32-bit extension of the industry standard microprocessor interface.
- Large virtual and physical address space — The chip supports four gigabytes (2^{32}) of virtual memory, and one gigabyte (2^{30}) of physical memory.
- High performance — At its maximum frequency, the chip achieves a 200 ns microcycle and a 400 ns I/O cycle.
- Single package — The chip is packaged in a standard 68-pin surface mounted chip carrier.

Key features supported by the DC337 FPU chip:

- Subset VAX data types — The chip supports the following subset of VAX data types: byte, word, longword, f floating, d_floating, and g_floating. The data type h_floating is not supported.
- Subset VAX instruction set — The chip implements a subset of the VAX floating point instruction set. (The remaining floating point instructions, except h_floating, are implemented in the CPU chip.) Accuracy for the EMOD and POLY instructions will meet VAX architectural standards.
- Integer multiply and divide acceleration — The chip supports signed integer multiply and unsigned integer divide.
- Simple external interface — The chip's external interface is straightforward and requires no external support chips.

- High performance — At its maximum frequency, the chip achieves a 100 ns microcycle and a 200 ns I/O cycle.
- Package — The chip is packaged in a 68-pin surface-mounted chip carrier.
- Fast instruction times — Table 2-1 lists typical instruction times for the FPU. Note that times may be faster or slower depending on the operands used in the calculation.

Table 2-1: FPU Instruction Times

Instruction	Single	Double
ADD[1]	2.0	2.6
MUL	2.6	4.2
DIV	3.7	6.1

[1]Digital FPU number uses operands separated by 11 in the exponent.

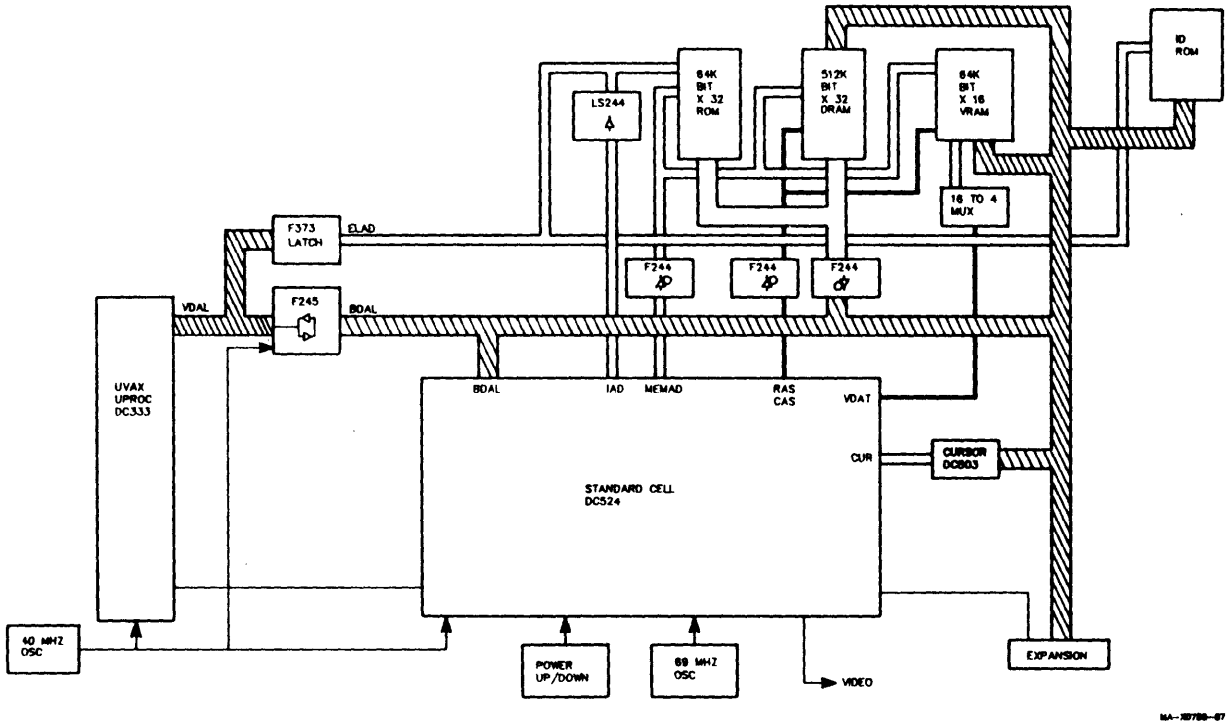
2.2 System Memory

The system memory consists of RAM and ROM memory located on the system module and also RAM memory located on the option memory module. Even though the optional RAM is not located on the system module, it is considered to be system memory. Figure 2-3 shows the functional block diagram of the system memory.

The system supports up to 16 megabytes of RAM (DRAM) memory, not including video RAM. The actual amount of RAM depends upon the option memory module installed. The data path to RAM memory is 32-bits wide. Data integrity is checked by a parity bit associated with each byte of memory. The RAM that is physically located on the system module contains 2048 kilobytes of memory.

The video RAM (VRAM) consists of 128 kilobytes of memory on the system module. It contains the video bitmap screen display information. The video bus caries the bitmap information from the VRAM through a multiplexer counter to the standard cell. The standard cell then generates the proper signals to display the video data, along with the cursor data, onto the video screen.

Figure 2-3: System Memory Functional Block Diagram



The optional memory module can contain up to 14,336 kilobytes of RAM, however, only 2048 kilobyte and 4096 kilobyte RAM option memory modules are supported. The system generates byte parity when writing to RAM memory and checks byte parity when reading from RAM memory.

The system module ROM contains 256k bytes of data that includes processor restart, diagnostic and console code, and I/O device drivers. The system ROM is addressed by the CPU chip over the ELAD bus and also by the standard cell over the MEMAD bus. The ROM outputs the data onto the MD bus which is buffered onto the BDAL bus and sent back to the CPU chip. The system ROM also contains interrupt vector routines that are addressed by the standard cell over the IAD bus.

The ThinWire Ethernet ID ROM on the system module contains 32 bytes of memory for a unique Ethernet network identification address for the system.

Each option module is required to have its own ROM memory that contains a standard signature to identify the option, as well as firmware initialization code and diagnostic code. This option ROM information is accessed through the memory option port.

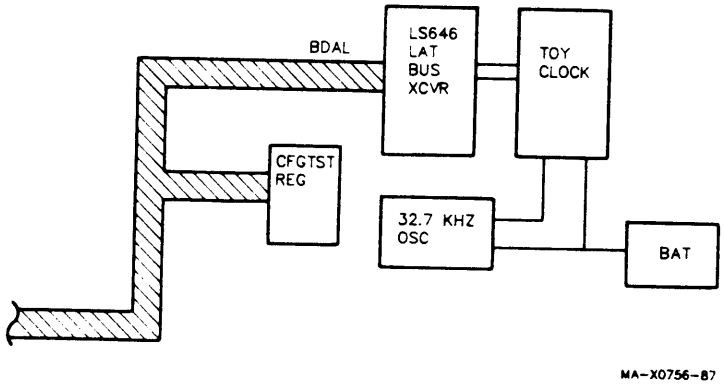
2.3 Time-Of-Year Clock

The time of year clock keeps the date, time of day, and 50 bytes of general purpose RAM. A 32.768 kHz time base oscillator provides the clock input and a rechargeable nickel-cadmium battery provides power to the chip and oscillator while system power is off. The TOY clock uses an LS646 transceiver to buffer and control the data and addresses to and from the CPU bus. Data from the TOY clock is used to determine the date and time during the power-up of the system. Within the 50 bytes of RAM are stored utilities such as the boot flags, boot device, halt action, and keyboard type as well as other volatile information. Figure 2-4 shows the functional block diagram of the TOY clock and also the configuration and test register.

A nickel-cadmium battery in the system box supplies power to the watch chip and its time base oscillator while system power is off. When starting from a fully charged condition, the battery maintains valid time and RAM data in the watch chip for a minimum of 100 hours. The battery recharges while system power is on.

Figure 2-4 also shows the configuration and test register. This register is an 8-bit register that contains system information such as whether the system is a VAXstation 2000 or a MicroVAX 2000, whether an option module is installed in the option slots, whether the BCC08 cable is connected to the printer port, and cursor chip test results.

Figure 2-4: TOY Clock Functional Block Diagram

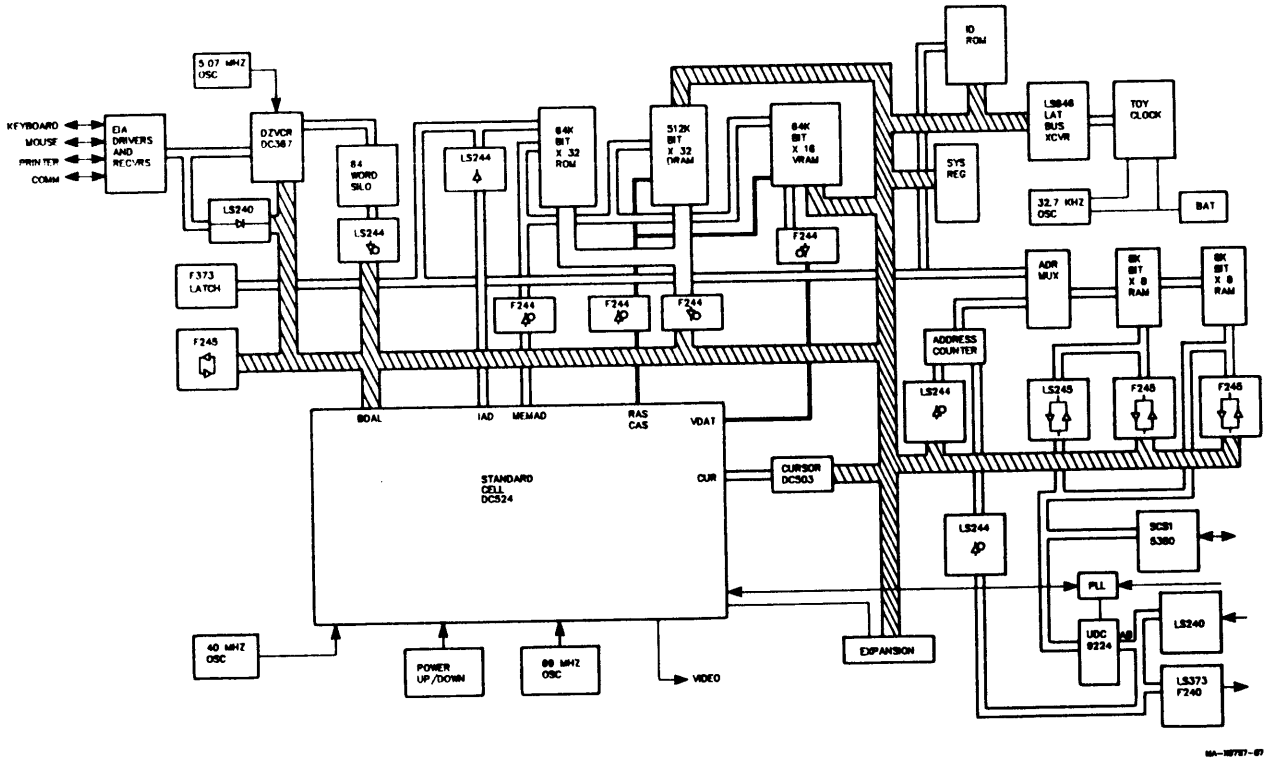


2.4 DC524 Standard Cell

The DC524 standard cell is the heart of the system. It controls the address decoding and the timing parameters for each device. It contains the interrupt controller, parity generation and checking, and all of the monitor timing circuitry internal to itself. The list below summarizes the functions of the standard cell and Figure 2-5 shows the functional block diagram of the standard cell.

- Power-up initialization
- Memory control
- Video control
- I/O control
- Disk control
- Tape control
- Parity generation and checking
- Interval timer interrupt generation
- Interrupt controller
- Monitor timing
- Chip test mode

Figure 2–5: DC524 Standard Cell Functional Block Diagram



2.5 DC503 Cursor Sprite Chip

The DC503 cursor sprite chip generates a cursor display on the video monitor. The cursor is generated from a two-plane memory array within the cursor chip. The cursor sprite chip receives commands over the BDAL bus for such things as cursor position, cursor pattern, and blanking of the cursor. The output of the cursor sprite chip is sent to the standard cell for inclusion in the video output signal to the monitor. Figure 2-6 shows the functional block diagram of the DC503 cursor sprite chip.

2.6 Serial Line Controller

The system module serial line controller handles four asynchronous serial lines. This controller is a DC367B gate array. Input characters from all four lines are buffered in a common 64-position silo. The silo is a true silo where a character drops through all 64 words in the silo before it is latched at the output. Only one line, the communication line, has full modem control signals. Figure 2-7 shows the functional block diagram of the serial line controller.

Figure 2-6: DC503 Cursor Sprite Chip Functional Block Diagram

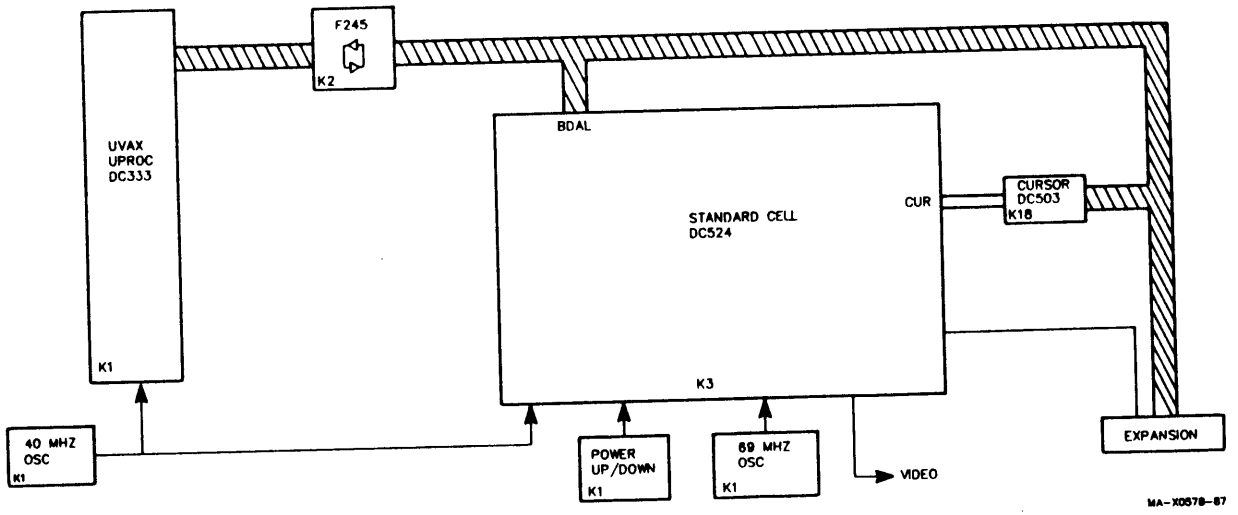
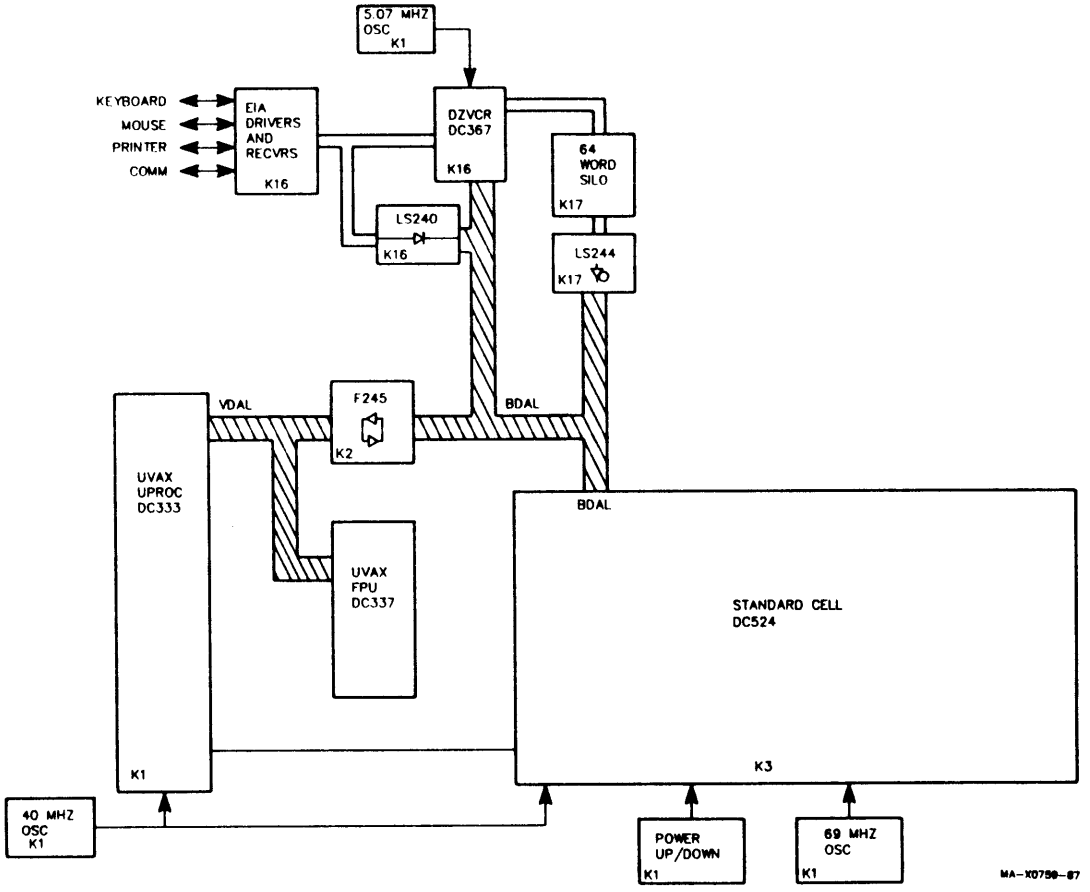


Figure 2-7: Serial Line Controller Functional Block Diagram



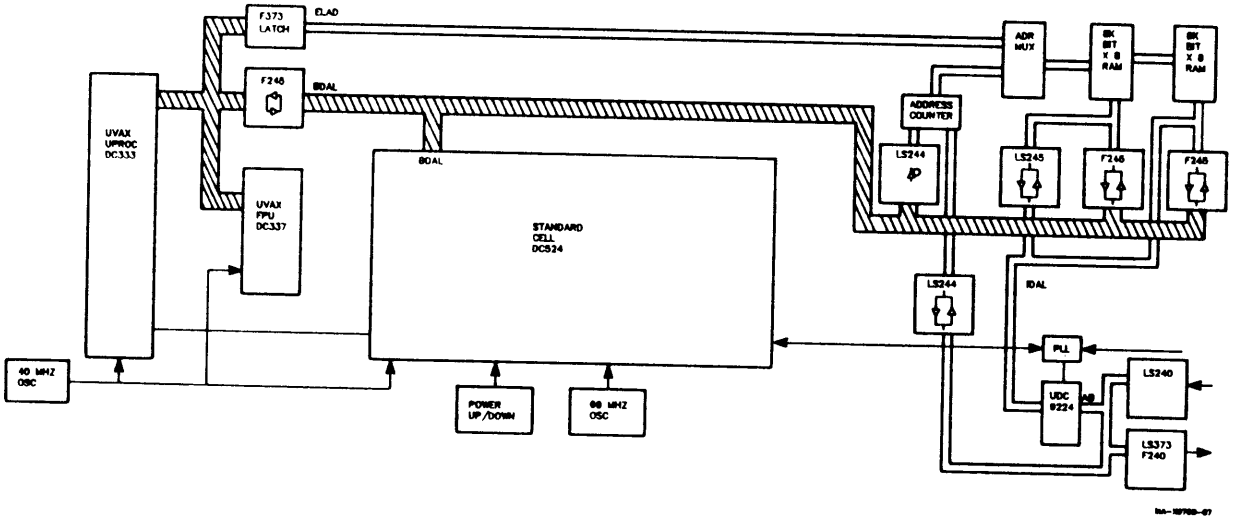
2.7 9224 Disk Controller

The disk controller supports both diskette drives (RX33) and ST506/412 hard disk drives (RD32 and RD53). The maximum configuration of the controller is one diskette drive and two hard disk drives. The controller is an HDC 9224 universal disk controller chip which uses a phase-locked loop (PLL) data recovery circuit, an address counter, and a 16-kilobyte dual port data buffer. Figure 2-8 shows the functional block diagram of the 9224 disk controller.

The disk data buffer is a 16-kilobyte block of RAM storage which is shared between the disk controller, the tape controller, and the CPU. This buffer uses two 8-kilobyte by 8-bit static RAM chips and is not included as part of the system module dynamic RAM. The disk and tape controller access the data buffer through the address counters. The address counters hold the data buffer address from the disk controller during normal RAM cycles as well as during DMA cycles. The disk data buffer is accessed by the CPU chip through the tri-state transceivers between the BDAL bus and the IDAL bus.

The phased locked loop (PLL) consists of a phase comparator and a voltage-controlled oscillator (VCO). The phase comparator is inside the standard cell. The VCO is a dual oscillator chip for both hard disk and floppy diskette data frequencies. The phased lock loop is used to control the frequency of the raw read data from the disks. The individual modified frequency modulation (MFM) pulses that are read from the disks are sensitive to speed variations and the value of the pulse (1 or 0) may be lost if the frequency of the data stream is not precise. The VCO allows the tracking of any variation of the data stream and sends feedback to the phase comparator to compensate the variation so the loop recovers the data and sends the disk controller a steady and reliable data stream.

Figure 2-8: 9224 Disk Controller Functional Block Diagram

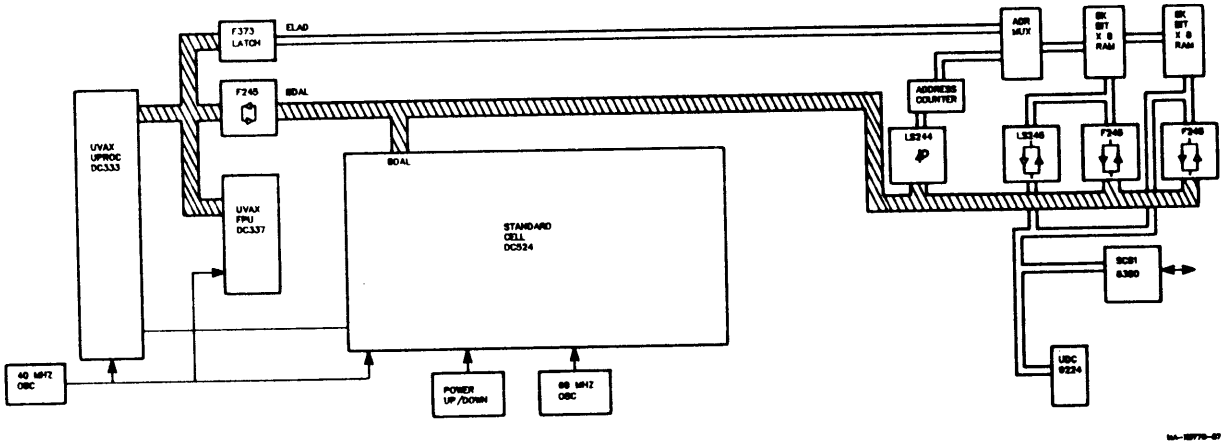


2.8 5380 Tape Controller

The tape controller is an NCR 5380 SCSI controller chip. It provides an ANSI Small Computer System Interface (SCSI) interface between the TZK50 tape controller in the tape expansion box and the data buffer on the system module. The tape controller is connected directly to the SCSI tape bus, which is port A on the expansion adapter, and it is also connected to the disk data buffer through the disk buffer data bus. The tape controller is controlled by the DC524 standard cell. Figure 2-9 shows the functional block diagram of the 5380 tape controller.

The SCSI interface is a bi-directional 8-bit wide bus to which up to eight devices can be attached. The system module is one of those devices, so up to seven additional devices can be attached. Devices may play one of two roles: initiator or target. An initiator originates an operation by sending a command to a specific target. A target performs an operation that is requested by an initiator. In this product, it is assumed that the system module is always an initiator and that all other SCSI devices attached to it are targets. Each device attached to the SCSI tape bus is identified by a unique device ID number in the range 0 through 7; the system module is normally 0.

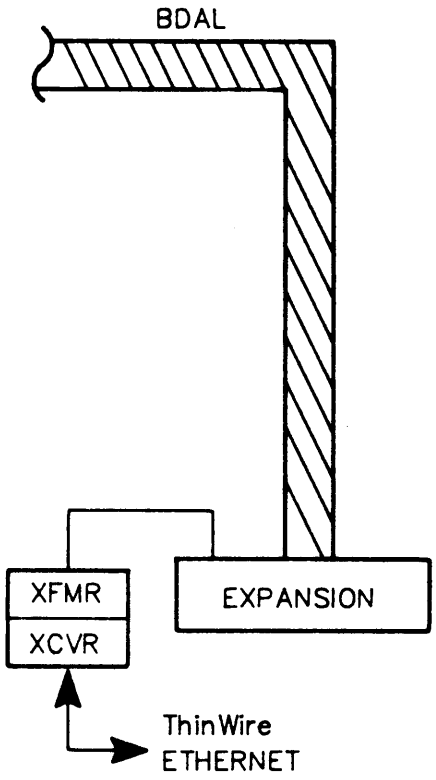
Figure 2-9: 5380 Tape Controller Functional Block Diagram



2.9 ThinWire Ethernet Circuits

The only portion of the ThinWire Ethernet network circuitry that is not on the Ethernet network option module is the transceiver circuitry. This transceiver circuitry is located in the upper right hand corner of the system module. It consists of the coaxial cable connector, the coaxial transceiver interface chip, and the isolation transformer. The coaxial transceiver interface (CTI) is used a the coaxial cable line driver and receiver for the ThinWire Ethernet local area network. The CTI contains a transmitter, receiver, collision detector, and a jabber timer. Figure 2-10 shows the functional block diagram of the ThinWire Ethernet circuits.

Figure 2-10: ThinWire Ethernet Circuits Functional Block Diagram



MA-X0771-87

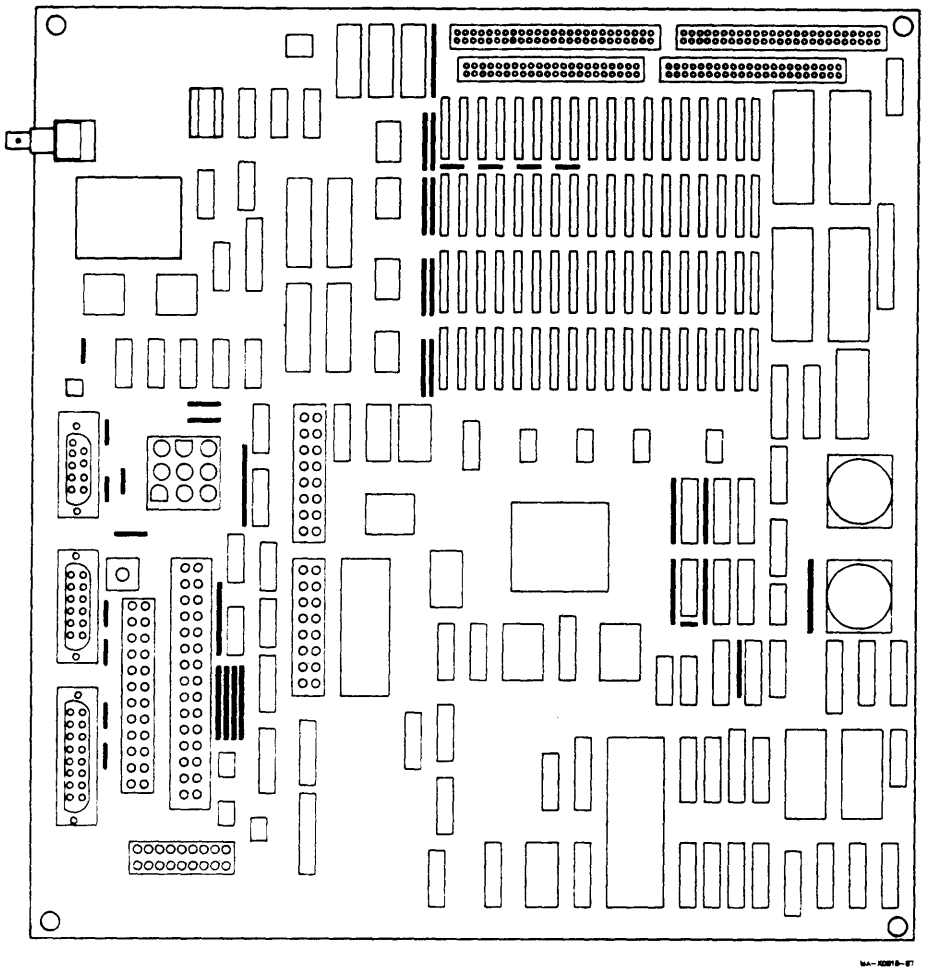
Chapter 3
VS410 System Module Detailed Description

3.1 Introduction

This chapter explains in detail the system module (see Figure 3–1). This chapter contains the following sections.

- Central processor
- ROM memory
- Time-of-Year clock
- DC524 standard cell
- DC503 cursor sprite chip
- Serial line controller
- 9224 disk controller
- 5380 tape controller
- ThinWire Ethernet circuits
- Miscellaneous registers
- VAXstation 2000 and MicroVAX 2000 system jumper configuration
- System module connector pinouts
- Power requirements

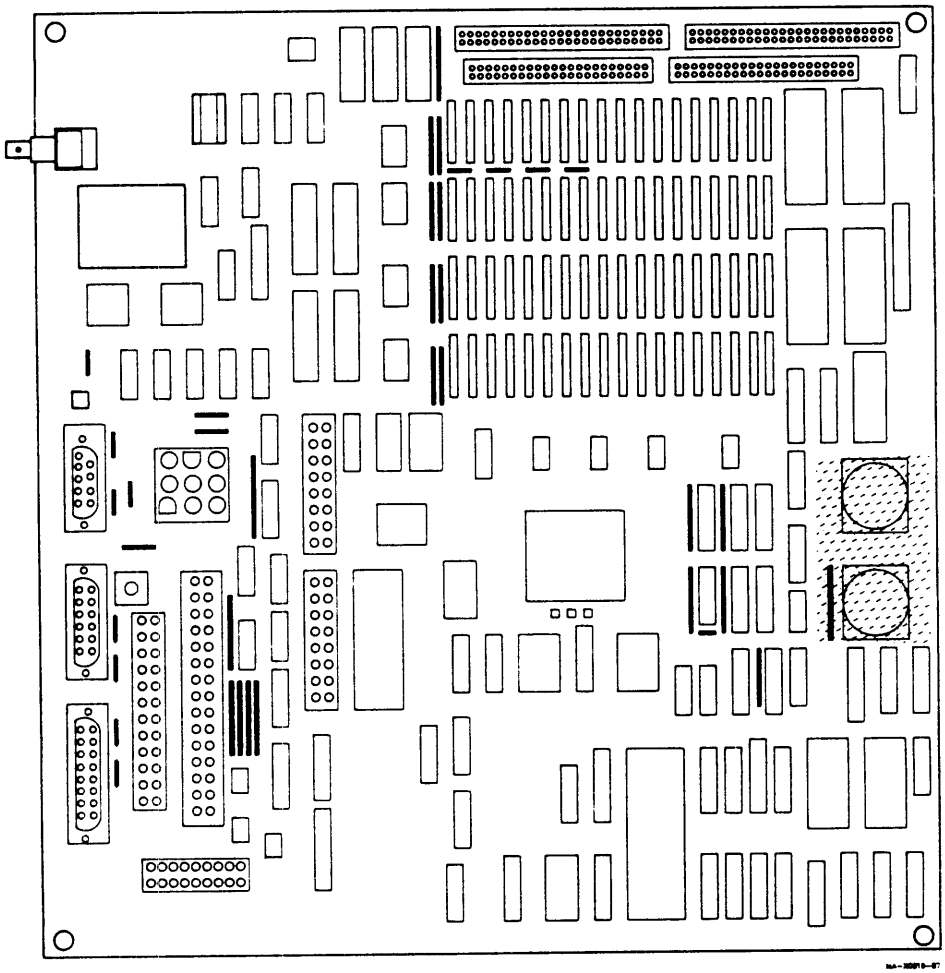
Figure 3–1: System Module



3.2 Central Processor

This section describes the CPU chip and the FPU chip in detail. See Figure 3-2.

Figure 3-2: Central Processor Unit (CPU)



3.2.1 DC333 CPU Chip Specifics

Figure 3-3 shows the pinout for the CPU chip. Table 3-1 lists the CPU pins and explains their functions.

Figure 3-3: CPU Chip Pinout

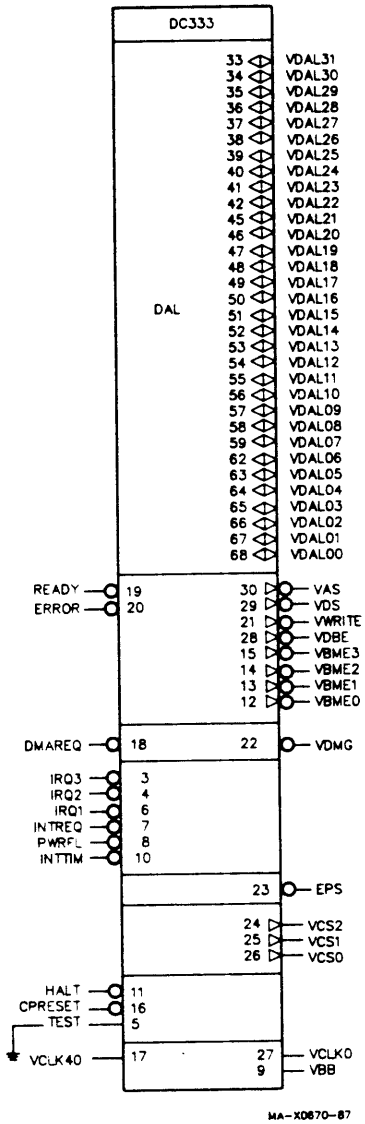


Table 3–1: CPU Chip Pin Functions

Pin	Signal	Description
CPU Data and Address Bus		
62:68 59:45 42:33	VDAL06:00 VDAL21:07 VDAL31:23	The data and address bus (VDAL 31:00) is a bidirectional time-multiplexed bus. During the first part of a CPU read cycle or CPU write cycle, VDAL31:30 indicate the length of the memory operand (00 = byte, 01 = word, 10 = longword, 11 = quadword), and VDAL29:00 contain the LONGWORD address of the memory operand (bit VDAL29 distinguishes memory space from I/O space). During the second part of a CPU read cycle or interrupt acknowledge cycle, VDAL31:00 is used to receive incoming information. During the second part of a CPU write cycle, VDAL31:00 is used to transmit outgoing information. During the first part of an interrupt acknowledge cycle, VDAL04:00 contain the IPL of the interrupt being acknowledged, VDAL29:05 contain 0s, and VDAL31:30 are 10. The VDAL bus is also used to exchange information with external processors such as the lance chip on the network interconnect option module.
Bus Control		
30	VAS	The address strobe signal provides timing and control information to the video and memory option ports. During a CPU read cycle, CPU write cycle, or interrupt acknowledge cycle, the chip asserts VAS L when the initial information on VDAL 31:00 is valid. The chip deasserts VAS L at the conclusion of the bus cycle.
29	VDS	The data strobe signal provides timing information for data transfers. During a CPU read cycle or interrupt acknowledge cycle, the chip asserts VDS L to indicate that VDAL 31:00 are free to receive incoming data, and deasserts VDS L to indicate that it has received and latched the incoming data. During a CPU write cycle, the chip asserts VDS L to indicate that VDAL 31:00 contain valid outgoing data, and deasserts VDS L to indicate the end of valid outgoing data.

Table 3–1 (Cont.): CPU Chip Pin Functions

Pin	Signal	Description
28	VDBE	The data buffer enable signal is used in conjunction with the VWRITE L signal to control external VDAL transceivers. The chip asserts VDBE L to enable the VDAL transceivers, and deasserts it to disable them.
21	VWRITE	The write signal specifies the direction of data transfer on the VDAL bus. If VWRITE L is asserted, then the chip is driving data onto the VDAL. If VWRITE L is not asserted, the chip is not driving data onto the VDAL. VWRITE L is valid when VAS L is asserted or EPS L is asserted.
20	ERROR	The DC524 standard cell asserts the bus error signal (ERROR L) to indicate abnormal termination of the current CPU read cycle, CPU write cycle, or interrupt acknowledge cycle. During a CPU read or CPU write cycle, this causes a machine check. During an instruction prefetch, this causes the prefetched data to be discarded. During an interrupt acknowledge cycle, ERROR L cancels the interrupt transaction. When the chip recognizes the assertion of ERROR L, it terminates the current bus cycle and proceeds. The DC524 standard cell then deasserts ERROR L.
19	READY	The DC524 standard cell asserts the ready signal (READY L) to indicate normal termination of the current CPU read cycle, CPU write cycle, or interrupt acknowledge cycle. During a CPU read cycle or interrupt acknowledge cycle, READY L indicates that the DC524 standard cell has placed the required input data on the VDAL bus. During a CPU write cycle, READY L indicates that the information is available on the VDAL bus. When the CPU chip recognizes the assertion of READY L, it terminates the current bus cycle and proceeds. The DC524 standard cell then deasserts READY L.
15:12	VBM3:0	The byte mask signals specify which bytes of the VDAL bus contain valid information during the second part of a CPU read cycle or CPU write cycle. If VBM3 L is asserted, then VDAL 31:24 contains valid data; if VBM2 L is asserted, then VDAL 23:16 contains valid data; if VBM1 L is asserted, then VDAL 15:8 contains valid data ; if VBM0 L is asserted, then VDAL 7:0 contains valid data.

Table 3–1 (Cont.): CPU Chip Pin Functions

Pin	Signal	Description
		During a CPU read cycle, the byte masks indicate which bytes of data must be placed on the VDAL; if this amounts to less than 32 bits, the other bytes of the VDAL are ignored. During a CPU write cycle, the byte masks specify which bytes of the VDAL bus contain valid data. During an interrupt acknowledge cycle, all four byte masks are asserted. VBM3:0 L are only valid when VAS L is asserted.
System Control		
26:24	VCS2:0	The control status lines, in conjunction with the VWRITE L signal, provide status about the current bus cycle. VCS2:0 are valid when VAS L or EPS L is asserted. (VCS2 is also used during the external processor protocol, see below). During a CPU read cycle, CPU write cycle, or interrupt acknowledge cycle (VAS L asserted), VWRITE L and VCS2:0 mean the following:

VWRITE	VCS2:0	Bus Cycle Type
H	LLL	reserved
H	LLH	reserved
H	LHL	reserved
H	LHH	interrupt acknowledge
H	HLL	read (I-stream)
H	HLH	read lock
H	HHL	read (D-stream, modify intent)
H	HHH	read (D-stream, no modify intent)
L	LLL	reserved
L	LLH	reserved
L	LHL	reserved
L	LHH	reserved

Table 3–1 (Cont.): CPU Chip Pin Functions

Pin	Signal	Description		
		L	HLL	reserved
		L	HLH	write unlock
		L	HHL	reserved
		L	HHH	write (D-stream)

During an external processor read cycle, external processor write cycle, or external processor response cycle (EPS L asserted), VCS2 is precharged and sustained high, and VWRITE L and VCS1:0 mean the following:

VWRITE	VCS1:0	Bus Cycle Type
H	LL	reserved
H	LH	read data
H	HL	reserved
H	HH	response enable
L	LL	write command (FPU)
L	LH	write data
L	HL	write command (non-FPU)
L	HH	reserved

16	CPRESET	The DC524 standard cell asserts the reset signal (RESET L) to force the CPU chip to its initial power-up state.

Table 3–1 (Cont.): CPU Chip Pin Functions

Pin	Signal	Description
11	HALT	Pressing the halt button or pressing the BREAK key on the diagnostic console asserts the halt signal (HALT L) to transfer control to console macrocode. At the conclusion of the current macroinstruction, the chip executes an external processor write cycle. During this cycle, VCS1:0 = 10 (non-FPU command) and VDAL05:00 = 111111. The chip then enters the restart process with the restart code = 2 (HALT L asserted). HALT L is edge-sensitive rather than level-sensitive, is sampled during every microcycle, and is synchronized internally.
Interrupt Control		
10	INTTIM	The interval timer signal (INTTIM L) allows the DC524 standard cell to signal an interval timer rollover to the chip. INTTIM L interrupts at IPL16 (SCB vector C0 hex). An interval timer interrupt is not acknowledged by the chip. INTTIM L is edge-sensitive rather than level-sensitive, is sampled during every microcycle, and is synchronized internally.
8	POWERFAIL	This signal is not used. It is pulled high by a pull-up resistor.
7	INTREQ	The interrupt request signal (IRQ0 L) from the DC524 standard cell allows several I/O devices to input a single interrupt request to the CPU chip at interrupt level IPL14. When taken, interrupt requests are acknowledged by an interrupt acknowledge cycle. IRQ0 L is level-sensitive, is sampled during every microcycle, and is synchronized internally.
6	IRQ1	This signal is not used. It is pulled high by a pull-up resistor.
4	IRQ2	This signal is not used. It is pulled high by a pull-up resistor.
3	IRQ3	This signal is not used. It is pulled high by a pull-up resistor.

Table 3–1 (Cont.): CPU Chip Pin Functions

Pin	Signal	Description
Direct Memory Access Control		
22	VDMG	The DMA grant signal (VDMG L) is asserted by the chip to grant control of the VDAL bus and related control signals to the DC524 standard cell and to the lance chip on the network interconnect option module. The chip floats (three-states) the VDAL bus and the related control signals. When the network interconnect module deasserts VDMAREQ L, the CPU chip responds by deasserting VDMG L and then starts the next bus cycle.
18	DMAREQ	The DMA request signal (DMAREQ L) is asserted by the lance chip on the network interconnect option module when it needs to take control of the VDAL bus and related control signals for DMA or other purposes. DMAREQ L is level-sensitive, is sampled during every microcycle, and is synchronized internally.
Miscellaneous		
27	VCLKO	This signal supplies a synchronized timing signal for other chips in the system. It oscillates at half the frequency of the 40 MHz clock input signal. The first rising edge of clock output following the deassertion of the reset signal begins the start of phase 1 of the CPU chip timing sequence.
23	EPS	The external processor strobe signal (EPS L) is used by the CPU chip to coordinate external processor transactions with the FPU chip.
17	CLKI	This input supplies a 40 MHz square wave clock timing to the CPU chip from an oscillator. Jumper W4 can be removed to disconnect the oscillator from the CPU chip for diagnostic purposes.
9	VBB	This pin is connected to the back bias generator. It can be used to test the function of the back bias generator or to supply back biasing during a diagnostic debug procedure.
5	Test	This signal is not used and is connected to ground.

3.2.1.1 CPU Bus Cycle Descriptions

The CPU chip supports eight types of bus cycles.

- Idle
- CPU read
- CPU write
- Interrupt acknowledge
- External processor read (status or data)
- External processor write (command or data)
- External processor response
- DMA

3.2.1.1.1 CPU Idle Cycle

An idle cycle requires four clock phases (nominally 200 ns). The VDAL bus is undefined. The bus control signals are unasserted.

3.2.1.1.2 CPU Read Cycle

In a CPU read cycle, the chip inputs information from main memory or I/O devices. A CPU read cycle requires a minimum of eight clock phases (nominally 400 ns) and may last longer, in increments of four clock phases (nominally 200 ns). The chip drives the physical longword address onto VDAL29:02. BM3:0 L and CS2:0 are asserted as required; WR L is unasserted. The chip asserts AS L, indicating that the physical address is valid. The chip then asserts DS L, indicating that the VDAL bus is free to receive incoming data. If no error occurs, external logic responds by placing the required data on VDAL31:00 and asserting RDY L. The chip then reads the data from the VDAL bus. If an error occurs, external logic responds by asserting ERR L. The chip ignores the data on VDAL31:00 in this case and, if the transaction is a data read, initiates a machine check. Finally, the chip deasserts AS L and DS L to end the CPU read bus cycle.

3.2.1.1.3 CPU Write Cycle

In a CPU write cycle, the chip outputs information to main memory or I/O devices. A CPU write cycle requires a minimum of eight clock phases (nominally 400 ns) and may last longer, in increments of four clock phases (nominally 200 ns). The chip drives the physical longword address onto VDAL29:02. BM3:0 L and CS2:0 are asserted as required, WR L is asserted. The chip asserts AS L, indicating that the physical address is valid. The chip then drives the output data onto VDAL31:00 and asserts DS L, indicating the data bus contains valid data. If no error occurs, external logic responds by reading the required data from the VDAL bus and asserting RDY L. If an error occurs, external logic responds by asserting ERR L, and the chip initiates a machine check. Finally, the chip deasserts AS L and DS L to end the CPU write bus cycle.

3.2.1.1.4 Interrupt Acknowledge Cycle

In an interrupt acknowledge cycle, the chip inputs a vector from an interrupting device. An interrupt acknowledge cycle requires a minimum of eight clock phases (nominally 400 ns) and may last longer, in increments of four clock phases (200 ns). The chip drives out the IPL of the interrupt being acknowledged on VDAL04:00 (IRQ0 L is IPL 14). VDAL29:05 are zero, VDAL31:30 are 10. BM3:0 L are all asserted, CS2:0 indicate an interrupt acknowledge cycle, and WR L is unasserted. The chip asserts AS L, indicating that the IPL level is valid. The chip then asserts DS L, indicating that the VDAL bus is free to receive the incoming vector. If no error occurs, external logic responds by placing the interrupt vector on VDAL09:02 and the normal processing flag on VDAL00 and asserting RDY L. The chip reads the vector from the VDAL bus. If an error occurs, external logic responds by asserting ERROR L. The chip ignores the data on the VDAL in this case and cancels the interrupt transaction. The chip deasserts AS L and DS L to end the interrupt acknowledge cycle.

The detailed timing of an interrupt acknowledge cycle is identical to a CPU read cycle.

3.2.1.1.5 External Processor Read Cycle

In an external processor read cycle, the chip inputs information (either status or data) from an external processor. An external processor read cycle lasts four clock phases (nominally 200 ns). The chip drives the cycle status onto CS1:0, precharges and sustains CS2 high, and asserts EPS L. The external processor responds by placing the required information onto the VDAL bus. The chip reads the information off the VDAL bus and deasserts EPS L and the external processor then removes its information from the VDAL to end the external processor read cycle.

3.2.1.1.6 External Processor Write Cycle

In an external processor write cycle, the chip outputs information (either command or data) to an external processor. An external processor write cycle lasts four clock phases (nominally 200 ns). The chip drives the cycle status onto CS1:0, precharges and sustains CS2 high, and asserts EPS L. The chip then places the outgoing information onto the VDAL bus and deasserts EPS L. The external processor responds to the deassertion of EPS L by reading the information off the VDAL bus to end the external processor write cycle.

3.2.1.1.7 External Processor Response Cycle

In an external processor response cycle, the chip inputs information (either status or data), and a completion or confirmation signal from an external processor. An external processor response cycle lasts four clock phases (nominally 200 ns). The chip drives the cycle status onto CS1:0, precharges and sustains CS2 high, and asserts EPS L. The external processor responds to the assertion of EPS L by placing the required information on the VDAL bus and, optionally, by driving CS2 low with an open drain driver. In any case, the chip deasserts EPS L. The external processor then removes its data from the VDAL, and stops driving CS2, if driven to end the external processor response cycle.

3.2.1.1.8 DMA Cycle

The chip can relinquish its control of the VDAL bus and related control signals upon request from the lance chip on the network interconnect option module for a DMA cycle. The lance chip requests control of the bus by asserting DMR L. At the conclusion of the current bus cycle, the CPU chip responds by floating (three-stating) VDAL31:00, AS L, DS L, WR L, and DBE L by driving BM3:0 L and CS2:0 high and by asserting DMG L (BM3:0 and CS2:0 are then floated also). The lance chip may now use the VDAL bus to transfer data. To return control of the VDAL bus to the CPU, the lance chip stops driving AS L, DBE L, and DS L, if driven, and deasserts DMR L. The chip responds by deasserting DMG L and starting the next bus cycle.

3.2.1.2 General Registers

There are sixteen general registers in the CPU chip. These registers contain 32 bits.

- Twelve general purpose registers (R0 - R11)
- One argument pointer register (R12,AP)
- One frame pointer (R13,FP)
- One stack pointer (R14,SP)
- One program counter (R15,PC)

3.2.1.3 Processor Status Longword (PSL) Register

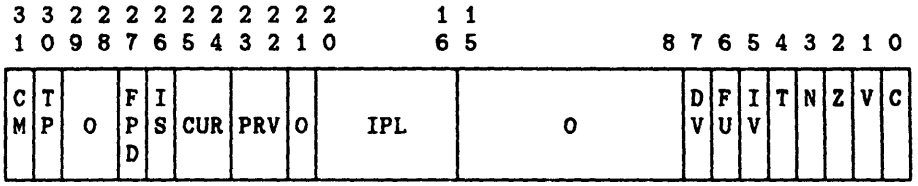
The PSL determines the execution state of the processor at any time. Figure 3-4 shows the format of the processor status longword.

3.2.1.4 Internal Processor Registers (IPR)

The internal processor registers are explicitly accessible only by the move to processor register (MTPR) and move from processor register (MFPR) instructions. Internal processor register space provides access to many types of CPU control and status registers such as the memory management base registers, parts of the process status longword, and the multiple stack pointers.

Table 3-2 enumerates the available processor registers and indicates how they are implemented in the VS410 system module. Registers that are not listed are reserved. Attempts to access a reserved register results in a reserved operand fault.

Figure 3-4: Processor Status Longword Register



Data Bit	Definition
31	CM—Compatibility mode
30	TP—Trace pending
29:28	0—Must be zero (0)
27	FPD—First part done
26	IS—Interrupt stack
25:24	CUR—Current mode
23:22	PRV—Previous mode
21	0—Must be zero (0)
20:16	IPL—Interrupt priority level
15:08	0—Must be zero (0)
07	DV—Decimal overflow trap enable
06	FU—Floating underflow fault enable
05	IV—Integer overflow trap enable
04	T—Trace enable
03	N—Negative condition code
02	Z—Zero condition code
01	V—Overflow condition code
00	C—Carry condition code

Table 3–2: Internal Processor Registers

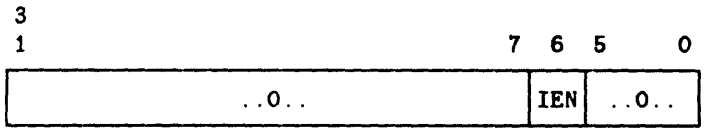
Number	Name	Description	Type	Note
0	KSP	Kernel stack pointer	R/W	1[1]
1	ESP	Executive stack pointer	R/W	1
2	SSP	Supervisor stack pointer	R/W	1
3	USP	User stack pointer	R/W	1
4	ISP	Interrupt stack pointer	R/W	1
8	P0BR	P0 base register	R/W	1
9	P0LR	P0 length register	R/W	1
10	P1BR	P1 base register	R/W	1
11	P1LR	P1 length register	R/W	1
12	SBR	System base register	R/W	1
13	SLR	System length register	R/W	1
16	PCBB	Process control block base	R/W	1
17	SCBB	System control block base	R/W	1
18	IPL	Interrupt priority level	R/W	1R[2]
19	ASTLVL	AST level	R/W	1R
20	SIRR	Software interrupt request	W	1
21	SISR	Software interrupt summary	R/W	1R
24	ICCS	Interval clock control	R/W	2R[3]
41	SAVISP	Console saved interrupt stack pointer	R/W	2
42	SAVPC	Console saved PC	R/W	2
43	SAVPSL	Console saved PSL	R/W	2
56	MAPEN	Memory management enable	R/W	1R
57	TBIA	Translation buffer invalidate all	W	1
58	TBIS	Translation buffer invalidate single	W	1
62	SID	System identification	R	1
63	TBCHK	Translation buffer check	W	1

[1] A 1 is Implemented as specified in the VAX Architecture Reference Manual (DEC STD 032).
[2] An R following the note number indicates that the register is cleared during power-up.
[3] A 2 is Implemented as specified in the MicroVAX CPU Chip Specification (A-PS-2120887-0-0).

3.2.1.4.1 Interval Clock Control and Status Register (ICCS)

The ICCS register controls the interval timer (INTTIM L) interrupt. The ICCS register is implemented as specified in the VAX architecture reference manual but it only contains a single bit to enable or disable the interval timer interrupt. Figure 3-5 shows the format of the ICCS register.

Figure 3-5: Interval Clock Control and Status Register (ICCS)

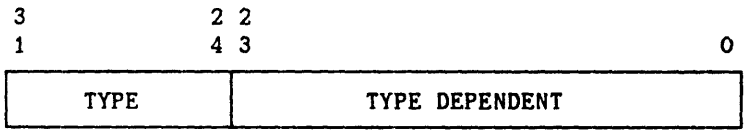


Data Bit	Definition
31:7	Not used. Read as zero (0). Ignored on writes.
IEN	Interrupt enable (bit 6). When this read/write bit is set, interval timer interrupts are disabled. At power-on, IEN is cleared.
5:0	Not used. Read as zero (0). Ignored on writes.

3.2.1.4.2 System Identification Register (SID)

The SID register (internal processor register 62, read-only) has the format shown in Figure 3-6. The TYPE field has the value 08h which identifies the processor as a DC333 MicroVAX CPU chip. The contents of the type dependent field are unpredictable.

Figure 3-6: System Identification Register (SID)



3.2.1.4.3 Console Saved Registers

The console saved registers (SAVISP, SAVPC, SAVPSL) record the value of the interrupt stack pointer, program counter (PC), and program status longword (PSL), respectively, at the time a chip restart occurs. See Section 3.2.6 for more information on the restart process.

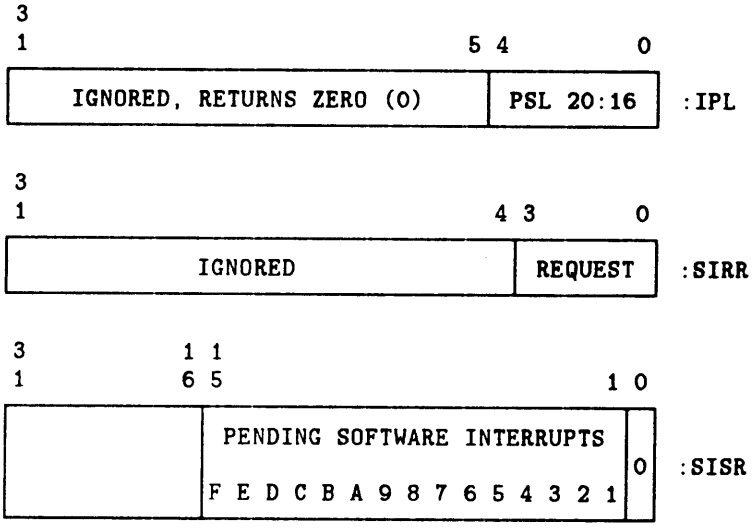
3.2.1.5 Interrupts and Exceptions

Both interrupts and exceptions divert program execution from its normal flow by pushing the processor status and program counter onto the stack and then beginning execution at the address found in one of the interrupt vectors in the system control block (SCB). An exception is typically handled by the current process (for example, an arithmetic overflow), while an interrupt typically transfers control outside the process (for example, an interrupt from an external hardware device).

3.2.1.5.1 Interrupts

The interrupt system is controller by the interrupt priority level register (IPL, internal processor register 18), the software interrupt request register (SIRR, internal process register 20), and the software interrupt summary register (SISR, internal process register 21). Figure 3-7 shows the format for all three of these registers.

Figure 3-7: Interrupt Control Registers (IPL, IRR, SISR)



3.2.1.5.1.1 Interval Timer Interrupts

An interval timer interrupt request is generated every 10 milliseconds by a signal on the INTTIM pin which is derived from the processor clock crystal. This interrupt is at IPL16h and uses interrupt vector 0C0h.

The interval clock control and status (ICCS) register (internal processor register 24, read/write) controls interval timer interrupts. Figure 3-5 shows the format of this register.

3.2.1.5.1.2 Device Interrupts

All interrupt requests from the system's I/O controllers are sent to the interrupt controller which ranks their priority and sends a single interrupt request to the CPU. The number of the interrupt is determined by the interrupt controller according to the identity of the requesting I/O controller. See Section 3.5.9.5 for a listing of the I/O controllers. Table 3-3 lists the external interrupts that are signalled to the CPU via one of three CPU chip pins.

Table 3-3: External Interrupts

CPU Pin	Interrupt
ERR	Machine check (bus error)
INTTIM	Interval timer interrupt; level 16h
IRQ0	Device interrupt; level 14h

The PWRFL, IRQ3, IRQ2, and IRQ1 interrupt pins on the CPU chip are not used and are held in the inactive state so the processor can never generate an interrupt on these lines.

3.2.1.5.2 Exceptions

The CPU chip recognizes six classes of exceptions, as follows.

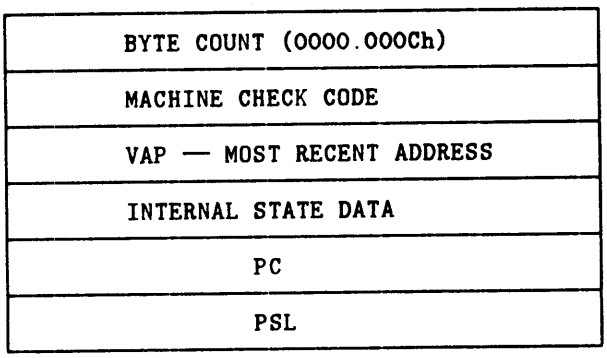
Exception Class	Instances
Arithmetic trap/fault	Integer overflow trap Integer divide by zero trap Subscript range trap Floating overflow fault Floating divide by zero fault Floating underflow fault
Memory management	Access control violation fault Translation not valid fault
Operand reference	Reserved addressing mode fault Reserved operand fault or abort
Instruction execution	Reserved privileged instruction fault Emulated instruction fault Extended function fault Breakpoint fault
Tracing	Trace trap
System failure	Memory read error abort Memory write error abort Kernal stack not valid abort Interrupt stack not valid abort Machine check abort

3.2.1.5.3 Machine Check Exceptions

A machine check exception results from either an internal CPU or FPU chip error or from the assertion of the ERR signal by external logic. The ERR signal is asserted when a RAM storage parity error is detected during a memory read cycle, which results in a machine check exception with a machine check code of either 80h or 81h. (Section 3.3.1.4 describes RAM storage parity checking.)

Figure 3-8 shows the parameters that are pushed onto the stack when a machine check exception occurs.

Figure 3-8: Machine Check Exception Parameters



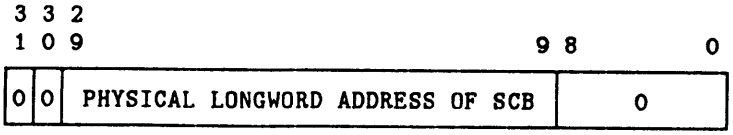
Byte count—The byte count is 0000.000Ch.
Machine check code (in HEX)—The machine check code is listed below.
VAP—Most recent virtual address. Not valid for machine check code 81h.
PC—Program Counter at the start of the current instruction.
PSL—Current contents of program status longword.

Code	Definition
1	Impossible microcode state (FSD)
2	Impossible microcode state (SSD)
3	Undefined FPU error code 0
4	Undefined FPU error code 7
5	Undefined memory management status TB miss
6	Undefined memory management status (M = 0)
7	Process PTE address in P0 space
8	Process PTE address in P1 space
9	Undefined interrupt ID code
80	Read bus error, VAP is virtual address
81	Read bus error, VAP is physical address

3.2.1.5.4 System Control Block

The system control block (SCB) is two physically-contiguous pages (1024 bytes) containing the vectors for servicing interrupts and exceptions. The first of its pages is pointed to by the system control block base register (SCBB, internal processor register 17). Figure 3-9 shows the format of the SCBB register. Table 3-4 lists the SCB format of the vectors used by this system.

Figure 3-9: System Control Block Base Register (SCBB)



Data Bit	Definition
31	Must be zero.
30	Must be zero.
29:9	Contains the physical longword address of the first page of the system control block.
8:0	Must be zero.

Table 3–4: System Control Block Format

Vectors	Vector Names	Vector Types
000	Unused	
004	Machine check[1]	Abort
008	Kernel stack invalid	Abort
00C	Power fail[2]	Interrupt
010	Resv/Priv. instruction	Fault
014	Customer resv. instr	Fault
018	Reserved operand	Fault/Abort
01C	Reserved addressing mode	Fault
020	Access control violation	Fault
024	Translation not valid	Fault
028	Trace pending	Fault
02C	Breakpoint instruction	Fault
030	Unused	
034	Arithmetic	Trap/Fault
038:03C	Unused	
040	CHMK	Trap
044	CHME	Trap
048	CHMS	Trap
04C	CHMU	Trap
050:080	Unused	
084:0BC	Software levels 1-15	Interrupt
0C0	Interval timer [3]	Interrupt

[1]Refer to Section 3.2.1.5.3.
[2]Refer to Section 3.2.1.5.
[3]Refer to Section 3.2.1.5.1.1.

Table 3–4 (Cont.): System Control Block Format

Vectors	Vector Names	Vector Types
0C4	Unused	
0C8	Emulation start	Fault
0CC	Emulation continue	Fault
0D0:0FC	Unused	
100:1FC	Adapter vectors	Interrupt
200:3FC	Device vectors [4]	Interrupt

[4]Refer to Section 3.5.9.

3.2.2 DC337 FPU Chip Specifics

Figure 3-10 shows the pinout for the FPU chip. Table 3-5 lists the FPU pins and explains their function.

Figure 3-10: DC337 FPU Chip Pinout

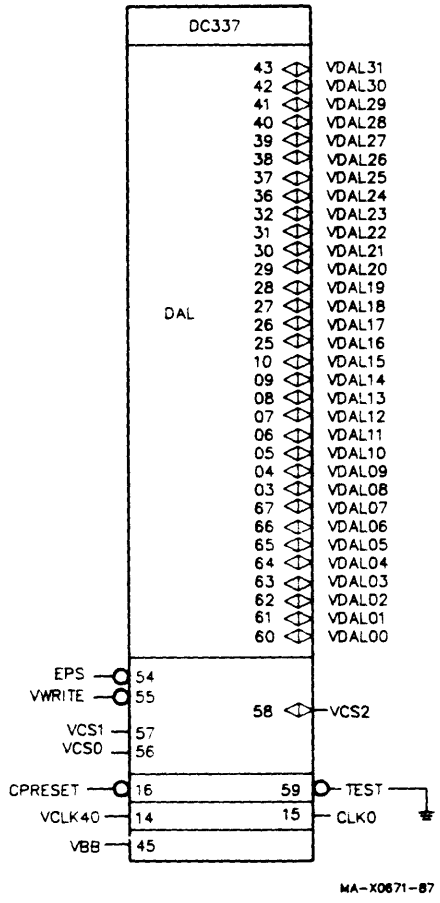


Table 3-5: DC337 FPU Chip Pin Functions

Pin	Signal	Description
FPU Data and Address Bus		
67:60 43:36 32:25 10:3	VDAL07:00 VDAL31:24 VDAL23:16 VDAL15:08	The data and address bus (VDAL31:00) is a bidirectional bus. It is used to exchange data between the CPU chip and the FPU chip. The CPU chip is always bus master.
FPU Control		
58:56	VCS2:0	The control status lines provide status about the current bus cycle. VCS1:0 are valid when EPS L is asserted. VCS1:0 are inputs which indicate the type of information being transferred. VCS2 is an open drain output which is active L when the current bus cycle is an external processor response enable and the FPU has completed the current commanded operation.

EPS	CS1:0	WR	Bus Cycle Type
L	LL	L	Write external processor command
L	LH	H	Read external processor data
L	LH	L	Write external processor data
L	HL	L	Command to external processors
L	HH	H	External processor response enable

55	VWRITE	The write signal is used by the CPU chip to indicate the direction of flow of data at the CPU. For the FPU, the write signal indicates that data is being transferred from the CPU.
54	EPS	The external processor strobe (EPSL) is used by the CPU chip to qualify all communication between the CPU chip and the FPU chip.

Table 3–5 (Cont.): DC337 FPU Chip Pin Functions

Pin	Signal	Description
Miscellaneous		
59	Test	This signal is not used and is connected to ground.
45	VBB	This pin is connected to the back bias generator. It can be used to test the function of the back bias generator or to supply back biasing during a diagnostic debug procedure.
16	CPRESET	The DC524 standard cell asserts the reset signal (CPRESET L) to force the chip to a known, initial state.
15	CLKO	This pin is not used.
14	CLKI	This input supplies a 40 MHz square wave clock timing to the FPU chip from an oscilator. This is the same clock that is sent to the CPU chip. Jumper W4 can be removed to disconnect the oscilator from the FPU chip for diagnostic purposes.

3.2.2.1 FPU Bus Cycle Descriptions

The FPU chip recognizes five types of bus cycles.

- FPU external processor command write
- *Other* external processor command write
- External processor read
- External processor write
- External processor response enable

3.2.2.1.1 FPU External Processor Command Write Cycle

In an FPU external processor command write cycle, the CPU chip outputs the instruction opcode to be read and executed by the FPU chip. An FPU external processor command cycle lasts eight FPU clock phases (nominally 200 ns). The CPU chip drives the cycle status onto CS1:0 and WRITE L. The CPU then loads the command onto the VDAL bus, and asserts EPS L. The FPU chip reads the data on the VDAL bus. The CPU chip deasserts EPS L and WRITE L to end the external processor command cycle.

3.2.2.1.2 *Other* External Processor Command Write

In an 'other' external processor command write cycle, the CPU chip outputs the instruction to be read and then executed by an external processor other than the FPU. If this is encountered, the FPU suspends operation of any instruction in progress and disables the output from responding to the CPU read cycle or response enable cycle until another FPU command is received.

3.2.2.1.3 External Processor Read Cycle

In an external processor read cycle, the CPU chip inputs information from the FPU chip. An external processor read cycle lasts eight FPU clock phases (nominally 200 ns). The CPU chip drives the cycle status onto CS1:0 and WRITE L, then asserts EPS L. The FPU chip responds by placing the required data onto the VDAL bus. The CPU chip reads the data off the VDAL bus and deasserts EPS L to end the external processor read cycle.

3.2.2.1.4 External Processor Write Cycle

In an external processor write cycle, the CPU chip outputs information to the FPU chip. An external processor write cycle lasts eight FPU clock phases (nominally 200 ns). The CPU chip drives the cycle status onto CS1:0 and asserts EPS L and WRITE L. The CPU chip then places the outgoing data on the VDAL bus and deasserts EPS L and WRITE L. The FPU chip responds to the deassertion of EPS L by reading the data off the VDAL bus.

3.2.2.1.5 External Processor Response Enable Cycle

In an external processor response enable cycle, the CPU tells the external processor that it is ready to accept a completion signal and that it controls the bus. The CPU drives the cycle status onto CS1:0 and WRITE L, then precharges and tristates the CS2 line. The FPU, when it has completed the current instruction, puts the status on the VDAL bus and pulls CS2 low with an open drain output device during the time the CPU asserts EPS L.

3.2.2.2 FPU/CPU Communications Protocol

The FPU/CPU communications protocol permits the CPU chip to communicate efficiently with the FPU chip. The general protocol for external processor communication follows these steps:

1. The CPU chip initiates the interaction by placing an FPU command on VDAL31:00, the FPU external processor command status code on CS1:0, and asserting WRITE L and pulsing EPS L. The FPU recognizes this as a command write cycle. Any instruction in-progress within the FPU is immediately aborted. The FPU decomposes the command to determine the operation to be performed and the number and size of the operands required.

2. The CPU chip next fetches the required operands and executes one or more external processor write cycles to transfer them to the FPU.
3. After the CPU chip has transferred the last operand, it asserts an external processor response enable code on the CS1:0 lines and pulses EPS L each microcycle that the CPU has control of the bus.
4. To signal non-completion of operations, the FPU does not affect CS2 when the external processor response enable code is on CS1:0 and EPS is low.
5. To signal completion of operations, the FPU asserts CS2 L when the external processor response enable code is on CS1:0 and EPS is low. At this same time, the FPU asserts the status of the just completed operation.
6. The CPU chip recognizes the CS2 L and reads the status information on the VDAL31:00 bus.
7. The CPU chip requests the status information again and is sent the status of the completed operation again.
8. The CPU chip next executes zero or more external processor read cycles to read the results of the computation, if any.

3.2.3 40 MHz CPU/FPU Clock

The processor clock input frequency is 40.0 MHz. This results in a microcycle time of 200 ns and an I/O cycle of 400 ns.

3.2.4 DMA Bus Access

The ThinWire Ethernet controller located on the network option module is the only controller in the system that can request DMA control over the system bus.

3.2.5 Memory Management

This section describes the management of the memory addressing space.

3.2.5.1 Virtual Memory Address Space

The CPU provides four gigabytes (2^{32}) of virtual memory address space. This virtual space is divided into two sections, process space and system space. Process space is further divided into a P0 region and a P1 region as shown in Figure 3-11. Process space (P0) virtual memory is mapped to physical memory by the P0 page table which is defined by the P0 base register (P0BR) and the P0 length register (P0LR). Process space (P1) virtual memory is mapped to physical memory by the P1 page table which is defined by the P1 base register (P1BR) and the P1 length register (P1LR). System space virtual memory is mapped to physical memory by the system page table which is defined by the system base register (SBR) and the system length register (SLR). The P0 region is accessed when address bits VDAL31:30 are both 0. The P1 region is accessed when address bit VDAL31 is 0 and VDAL30 is a 1. The system space is accessed when address bit VDAL31 is a 1 and VDAL30 is a 0.

3.2.5.2 Physical Memory Address Space

The CPU provides one gigabyte (2^{30}) of physical memory address space. Figure 3-12 shows the physical memory address space.

Figure 3-11: Virtual Memory Address Space

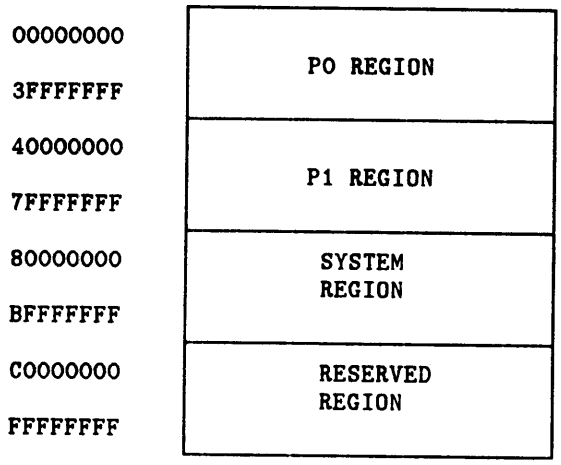
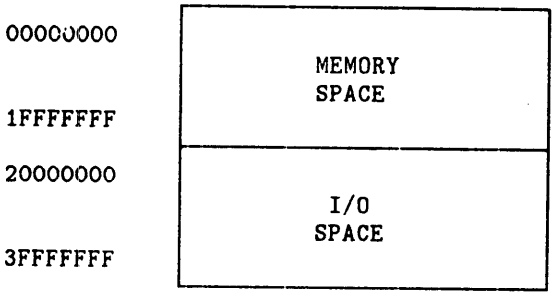


Figure 3-12: Physical Memory Address Space



3.2.5.3 Memory Management Control Registers

Memory management is controlled by three internal processor registers. These registers are the memory management enable (MAPEN), translation buffer invalidate single (TBIS), and translation buffer invalidate all (TBIA). MAPEN contains one bit which enables memory management (MAPEN0) as shown in Figure 3-13. TBIS controls translation buffer invalidation (Figure 3-14). Writing a virtual address into TBIS invalidates any entry which maps that virtual address. TBIA also controls translation buffer invalidation (Figure 3-15). Writing a zero into TBIA invalidates the entire translation buffer.

Figure 3-13: Memory Management (Mapping) Enable Register (MAPEN)

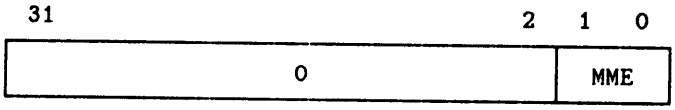


Figure 3-14: Translation Buffer Invalidate Single Register (TBIS)

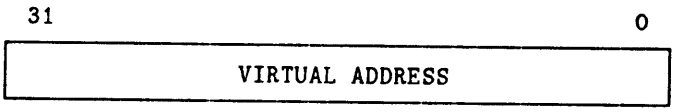
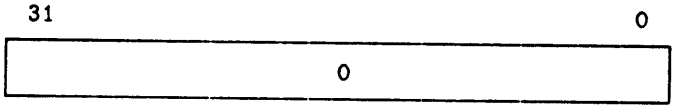


Figure 3-15: Translation Buffer Invalidate All Register (TBIA)



3.2.5.4 System Space Address Translation

A virtual address with bits 31:30 = 2 is an address in the system virtual address space. Refer to Figure 3-16. System virtual address space is mapped by the system page table (SPT), which is defined by the system base register (SBR) and the system length register (SLR). The SBR contains the physical address of the SPT. The SLR contains the size of the SPT in longwords, that is, the number of page table entries (PTEs). The PTE addressed by the SBR maps the first page of system virtual address space, that is, virtual byte address 80000000 (hex).

3.2.5.5 Processor Space Address Translation

A virtual address with bit 31 = 0 is an address in the process virtual address space. Process space is divided into two equally sized, separately mapped regions. If virtual address bit 30 = 0, the address is in region P0. If virtual address bit 30 = 1, the address is in region P1.

3.2.5.5.1 P0 Region Address Translation

Refer to Figure 3-17. The P0 region of the address space is mapped by the P0 page table (P0PT), which is defined by the P0 base register (P0BR) and the P0 length register (P0LR). The P0BR contains the system virtual address of the P0PT. the P0LR contains the size of the P0PT in longwords, that is, the number of PTEs. The PTE addressed by the P0BR maps the first page of the P0 region of the virtual address space, that is, virtual byte address 0.

3.2.5.5.2 P1 Region Address Translation

Refer to Figure 3-18. The P1 region of the address space is mapped by the P1 page table (P1PT), which is defined by the P1 base register (P1BR) and the P1 length register (P1LR). Because P1 space grows toward smaller addresses, and because a consistent hardware interpretation of the base and length registers is desirable, P1BR and P1LR describe the portion of P1 space that is not accessible. Note that P1LR contains the number of nonexistent PTEs. P1BR contains the virtual address of what would be the PTE for the first part of P1, that is, virtual byte address 40000000 (hex). The address in P1BR is not necessarily a valid physical address, but all the addresses of PTEs must be valid physical addresses.

Figure 3-16: System Space Virtual to Physical Address Translation

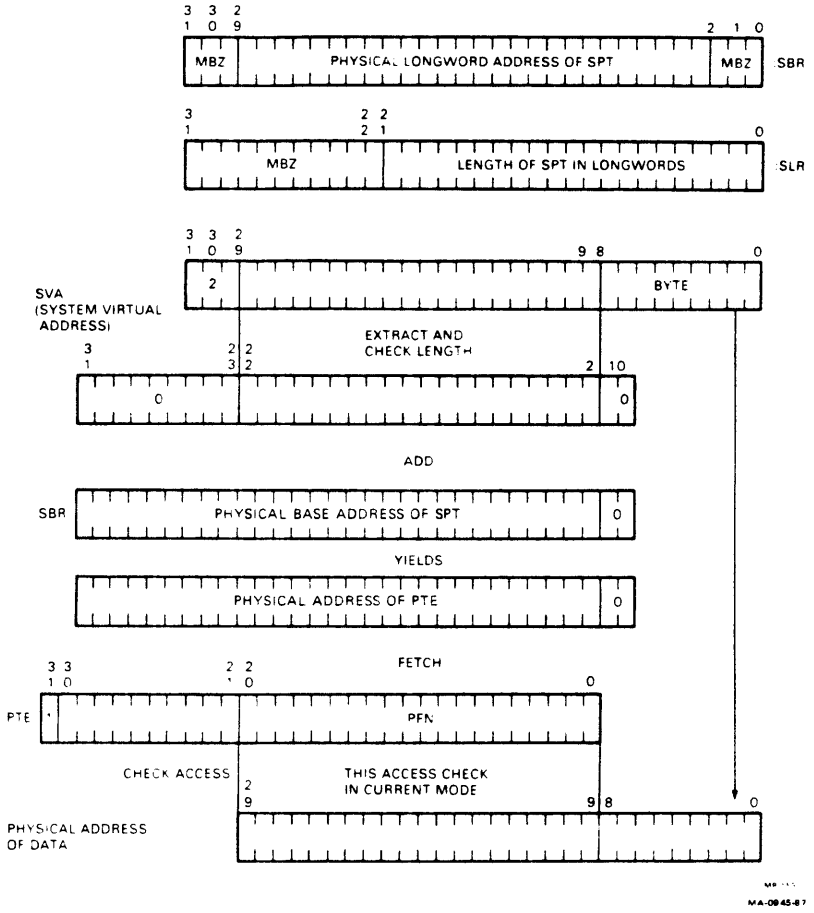


Figure 3-17: P0 Virtual to Physical Address Translation

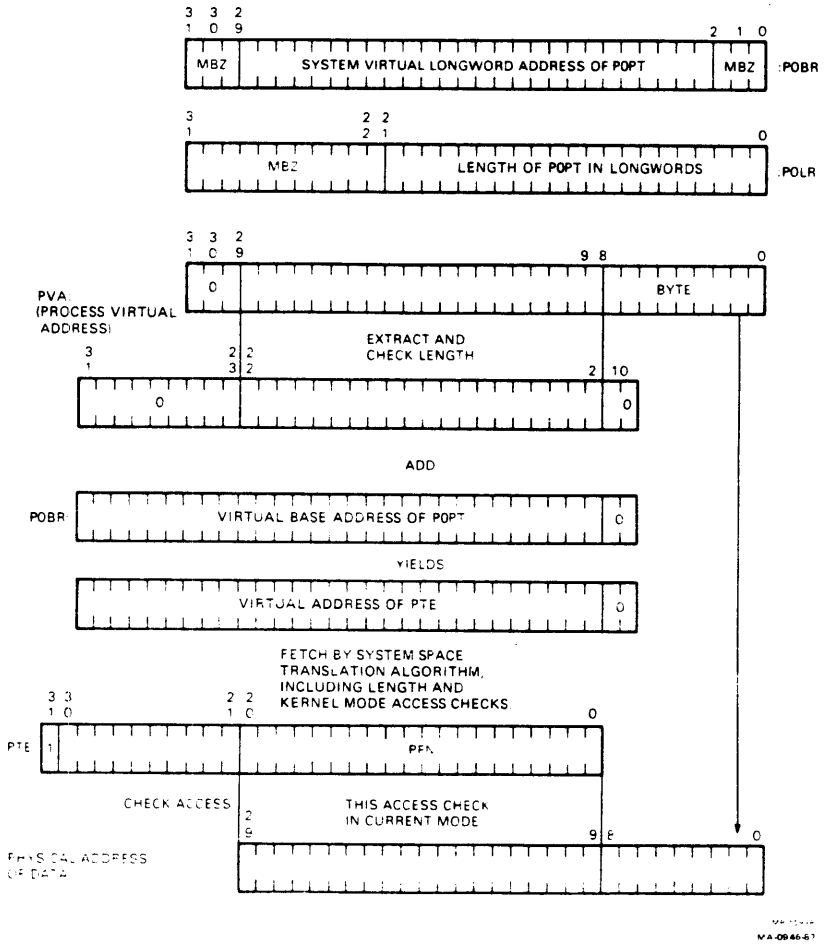
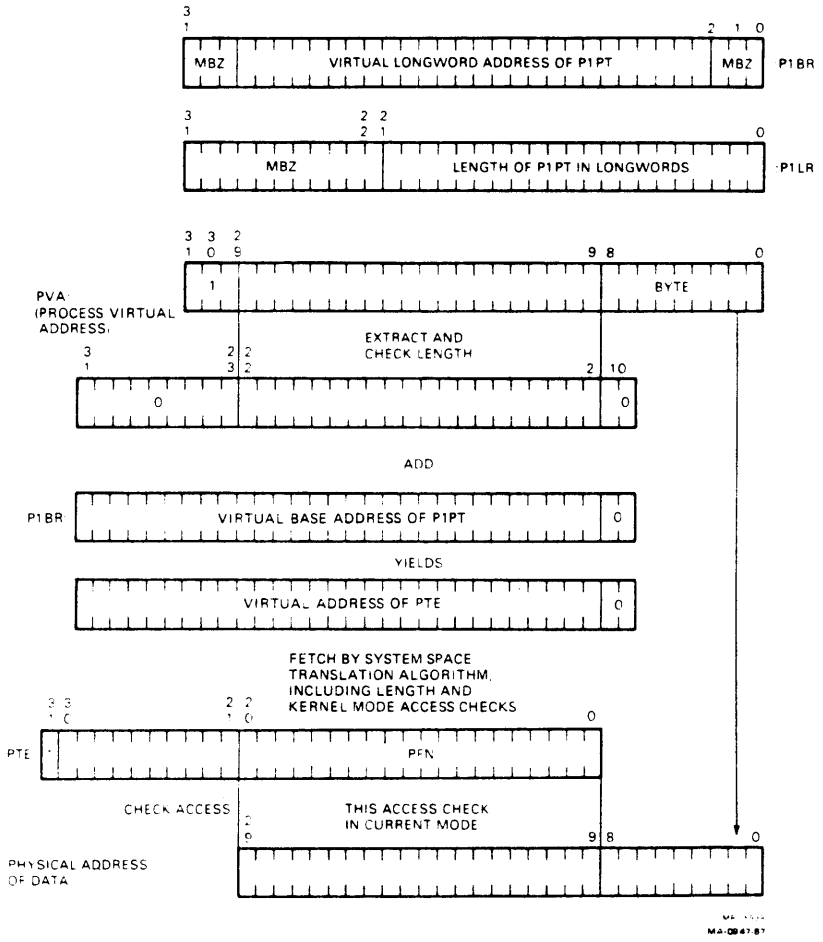


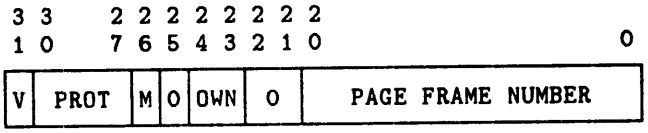
Figure 3-18: P1 Virtual to Physical Address Translation



3.2.5.6 Page Table Entry

The format of a valid PTE is shown in Figure 3-19. If bit 31 (the V bit) is clear, the format of the remaining bits is not examined by the hardware.

Figure 3-19: Page Table Entry (PTE)



Data Bit	Definition
V	Valid bit (bit 31). This bit must be set.
PROT	Protection code (bits 30:27)
M	Modify bit (bit 26)
25	Must be zero
OWN	Owner bits (bits 24:23)
20:0	Page frame number

3.2.6 Processor Restarts

When the CPU receives a RESET or HALT or detects severe corruption of its operating environment, it performs a restart process. This restart process saves some of the contents of the internal processor registers (SAVISP, SAVPC, and SAVPSL), changes the CPU to unmapped memory mode, and begins program execution in the system ROM at address 2004.0000. Bits 14:8 of SAVPSL contain a restart code which indicates the cause of the restart. The restart codes (in hex) are listed below.

Restart	Definition
2	HALT asserted (See Section 3.2.6.2 below.)
3	Power on
4	Interrupt stack not valid during exception
5	Machine check during machine check, or kernel stack not valid exception
6	HALT instruction executed in kernel mode
7	SCB vector bits 1:0 = 11
8	SCB vector bits 1:0 = 10
A	CHMx executed while on interrupt stack
10	ACV or TNV during machine check exception
11	ACV or TNV during kernel stack not valid exception

The restart process sets the state of the chip as follows.

Register	Contents
SAVISP	Saved interrupt stack pointer
SAVPC	Saved PC
SAVPSL	Saved PSL bits 31:16 and 7:0 in bits 31:16 and 7:0 Saved MAPEN 0 in bit 15 Saved restart code in bits 14:8
SP	Stack pointer at time of restart (not the stack pointer specified by bits 26:24 in the PSL)
PSL	041F 0000 (hex)
PC	2004 0000 (hex)
MAPEN	0
SISR	0 (power on only)
ASTLVL	4 (power on only)
ICCS	0 (power on only)
	All other registers are undefined.

3.2.6.1 Power-On Restart

The system performs a power-on restart whenever power is switched on. The CPU's RESET pin (CPRESET L signal) is held low by the standard cell during the power-on initialization of the system. The standard cell holds CPRESET low to ensure that the CPU sees an adequate number of clock cycles while in the reset state. Once initialization is complete, the standard cell allows CPRESET to go high. The CPU performs a power-on restart with a restart code of 3 in bits 14:8 of SAVPSL.

3.2.6.2 HALT Restarts

The system performs a HALT restart when the CPU's HALT pin (HALT L signal) goes low. The HALT L signal drops low whenever the operator does one of the two following things.

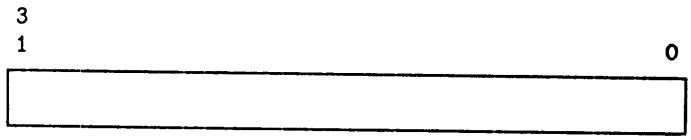
1. Pressing the operator's halt button on the rear of the system box.
2. Pressing the BREAK key on the terminal connected to serial line 3 (printer connector) with a BCC08 console cable.

Upon receiving the HALT L signal, the CPU enters console mode. The operator can then examine and alter storage, run diagnostics, or initiate a system bootstrap. The CPU performs a HALT restart with a restart code of 2 in bits 14:8 SAVPSL.

3.2.6.3 HALT Code Register (HLTCOD)

The halt code register (HLTCOD) is a read/write longword register at physical address 2008.0000. It is intended for use by the ROM-resident firmware program which handles a processor restart. This program moves internal processor register SAVISP to HLTCOD so that the restart code can be extracted without accessing any of the processor's general registers or any RAM locations.

Figure 3-20: Halt Code Register (HLTCOD)

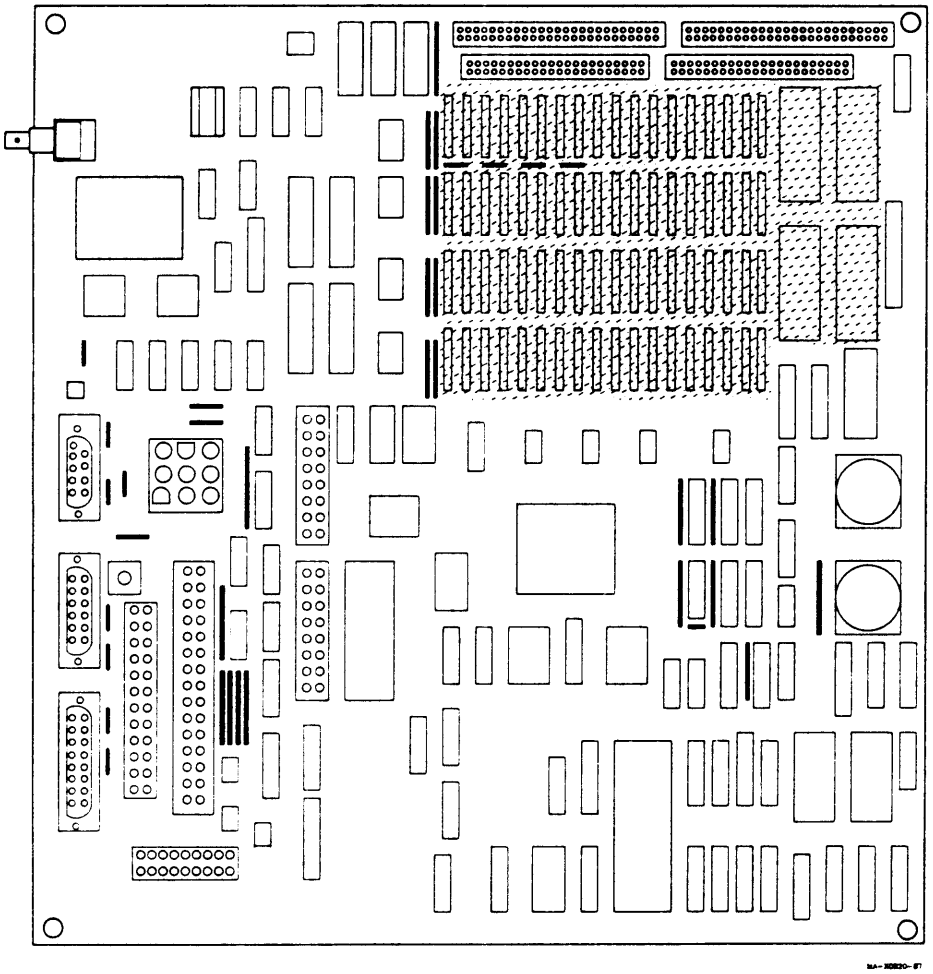


NOTE: *There is an interaction between the HLTCOD register and the SCD_CNT register (described in Section 3.9.4.2). The contents of HLTCOD must be 0 whenever a program attempts to read the contents of SCD_CNT; otherwise the value received may be in error. The contents of HLTCOD do not affect program writes to SCD_CNT and do not affect actual DMA operation.*

3.3 System Memory

This section describes system memory (Figure 3-21), including the system RAM, video RAM, and ROM in detail. Chapter 4 describes the option memory module in detail.

Figure 3-21: System Memory



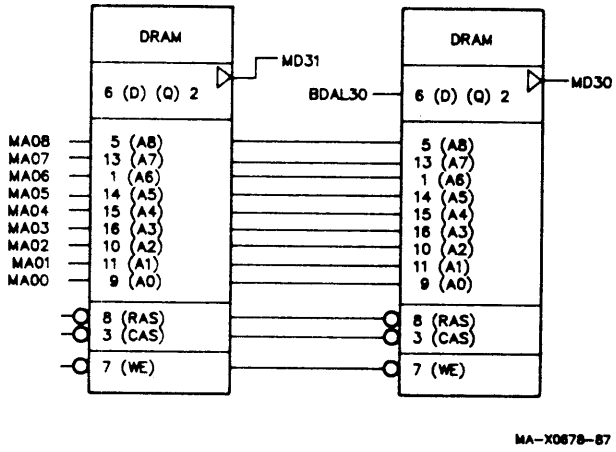
3.3.1 RAM Memory

The system supports up to 16 megabytes of RAM memory. The actual amount of RAM depends upon the option memory module installed. The data path to RAM memory is 32 bits wide. Data integrity is checked by a parity bit associated with each byte of memory.

3.3.1.1 System Module RAM

The system module contains 2048 Kbytes of RAM which occupies physical addresses 0000.0000 through 001F.FFFF. The RAM is stored in two banks of 32 individual 256Kx1 chips which are in a zig-zag in line package (zip packs). The chips have one data input line and one data output line which connects to one data bit on the BDAL31:00 bus. When the chips are properly addressed and selected, a single data bit from the BDAL31:00 bus is written to or read from the address location on each of the 32 chips. The system also provides memory refresh to the zip packs. Figure 3-22 shows the block diagram of two zip packs in bank 0 for RAM bits 31:30.

Figure 3-22: RAM Zip Packs Block Diagram



There are 262144 storage cells in each zip pack which are arranged in an array of 256 words (rows) by 1024 bits (columns). Addressing these cells is done in two steps. The first step selects the row address location and the second step selects the column address location. During the first step, an address from the DC524 standard cell is put on the memory address bus and the RAS control signal is asserted. One of the 256 rows is now selected. Before the second step is started, the DC524 standard cell floats the memory address bus to clear the row address. During the second step, another address from the DC524 standard cell is put on the memory address bus and the CAS control signal is asserted. The row and column addresses have now uniquely defined a single storage cell for writing to or reading from. If the system is writing to RAM (see Figure 3-23), the final step asserts the WE control signal and puts the data bit onto the CPU data bus which is then stored in the zip pack chip. If the system is reading from RAM (see Figure 3-24), the final step outputs the value of the addressed data bit onto the CPU data bus. This process occurs simultaneously on all 32 zip packs to get the 32-bit wide data bus. There are also four zip packs (one for each byte) for parity.

Figure 3-23: Data In (Write) Memory Timing Cycle

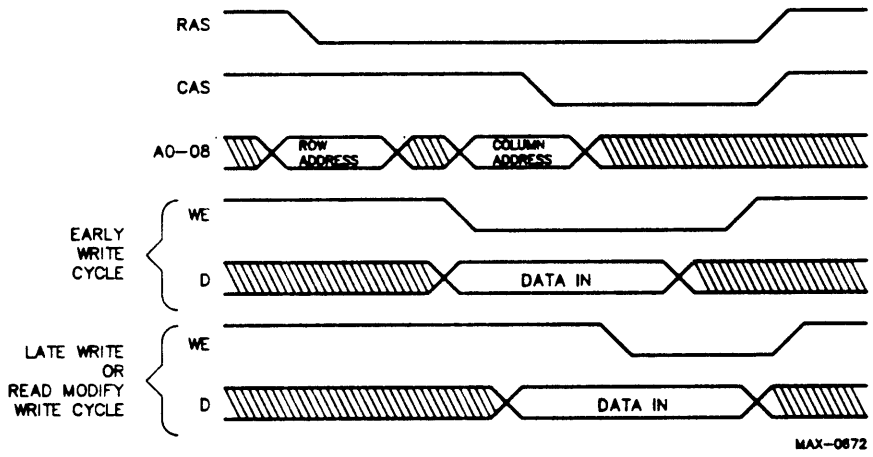
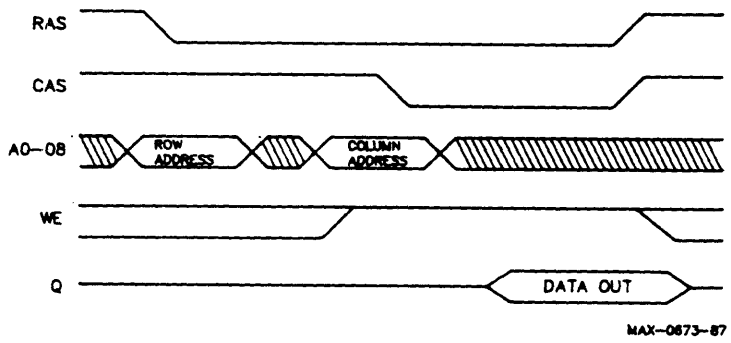


Figure 3-24: Data Out (Read) Memory Timing Cycle



3.3.1.2 Video RAM

The video RAM (VRAM) is a dual port 64kx4 RAM. There are four VRAMs on the system module. Addressing these VRAMs is done similar to the DRAMs so addressing is not covered here. The VRAMs each have four 1024-bit shift registers on their output which contain four rows of video data that is put onto the video bus (VID15:00). These shift registers are loaded once every four scan lines at the start of a refresh cycle by raising DT/OE and asserting VRAS. This allows the current row of video data to drop into the shift registers. There are two counters inside the standard cell; one is a refresh counter that keeps track of the row address for the refresh and the other is for keeping track of where the system is in the VRAM for the correct dots.

The video bus is then multiplexed down by a johnson style counter to four lines which go to the standard cell. The standard cell then generates the proper signals to display the video data, along with the cursor data, onto the video screen.

3.3.1.3 Option Module RAM

A memory option module can contain up to 14,336 Kbytes of RAM which begins at physical address 0020.0000 and continues through contiguous addresses to the capacity of the module. The presence and size of a memory option module can be determined by reading the MTYPE bits of the configuration and test register (CFGTST) (see Section 3.11.2).

3.3.1.4 Memory Parity Checking

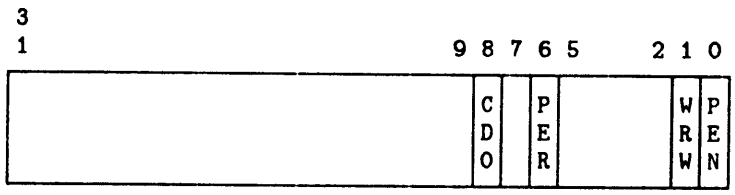
The system generates byte parity when writing to RAM memory and checks byte parity when reading from RAM memory. Parity checking applies both to CPU accesses and to DMA accesses generated by the network interconnect option. Only those bytes selected by the processor byte mask are affected and checked. Two I/O registers are associated with the parity system: the memory system error register (MSER) and the memory error address register (MEAR).

Parity generation and checking is active only in the physical address range 0000.0000 through 00FF.FFFF. Any read reference within this range to uninstalled memory may result in a parity error. References to uninstalled memory or nonexistent devices, in the physical address range 0100.0000 through 3FFF.FFFF, return unpredictable data upon reading and ignore this data upon writing. No parity error ever results from references in this range.

3.3.1.5 Memory System Error Register (MSER)

The memory system error register (MSER) is a longword at physical address 2008.0004 that controls the parity generation and checking logic and indicates when a parity error has been detected. Figure 3-25 shows the MSER register.

Figure 3-25: Memory System Error Register (MSER)



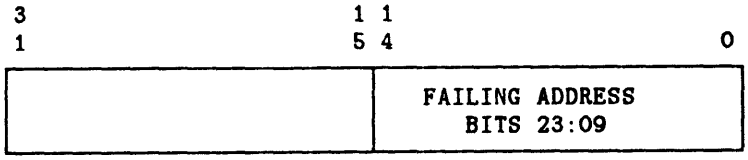
Data Bit	Definition
31:9	Not used. Read as 0.
CD0	Memory Code 0 (bit 8). Read-only. Duplicates the state of the PER bit (see below).
7	Not used. read as 0.
PER	Parity Error (bit 6). When parity error detection is enabled (bit PEN of this register is set), PER is set at the end of any CPU or DMA read access to any byte in RAM memory which contains incorrect parity. The first assertion of PER captures the number of the page containing the incorrect byte in the MEAR register and asserts the ERR signal to the CPU and to the network controller option module. Once the PER signal has been set, the next CPU data stream (not instruction stream) read bus cycle or DMA read bus cycle clears ERR again. If that next bus cycle was issued by the CPU, then a machine check exception occurs, regardless of whether the parity error was detected during a CPU or DMA bus cycle. If, however, the next bus read cycle following the detection of a parity error is a DMA read bus cycle (that is, two consecutive DMA read cycles without releasing the bus to the CPU), the ERR signal is cleared and the CPU does not see or generate a machine check (even though PER is still set). In order that such a parity error not pass undetected, the network controller option must monitor the ERR signal during DMA transfers and must inform its driver software when it detects a parity error. In summary, when a parity error is detected and PER is not already set, then it is detected by either the CPU or the network option controller, but not both. Further, the CPU may see parity errors which occurred during either CPU or DMA cycles, but the network controller sees only parity errors which occurred during DMA cycles. PER is cleared by writing to the MSER register with a 1 in the PER bit position. writing a zero does not affect PER. PER is also cleared upon power-on. Software which finds PER set must take appropriate action (possibly using the contents of MEAR), and then clear PER. Until PER is cleared again, no additional parity errors will be detected.
5:2	Not used. Read as 0.
WRW	Write Wrong Parity (bit 1). This read/write bit, when set, causes incorrect parity to be written by write accesses to RAM memory (i.e. a parity value which when read will signal a parity error). This bit is cleared during power-up and must be clear for normal operation.

Data Bit	Definition
PEN	Parity Enable (bit 0). This read/write bit must be set to enable the detection of incorrect parity to set the PER bit. When PEN is clear, parity errors are not recorded and have no effect on system operation. This bit is cleared during power-up.

3.3.1.6 Memory Error Address Register (MEAR)

The memory error address register (MEAR) is a longword at physical address 2008.0008 which captures part of the address of a byte that has incorrect parity. Figure 3-26 shows the MEAR register.

Figure 3-26: Memory Error Address Register (MEAR)



Data Bit	Definition
31:15	Not used. Read as 0.
14:0	Failing address. These read-only bits record bits 23:09 (the page number) of the physical address of the failing byte when a parity error is detected. They are latched at the same time that bit PER of the MSER register is set and they are valid only when PER is set. In the event that multiple parity errors occur before PER is cleared, MEAR contains the address associated with the first error (that is, the error which changed PER from 0 to 1). If the MEAR register is read while PER is clear, bits 23:09 of the MEAR register's own address are returned, that is, a value of 0000.0400.

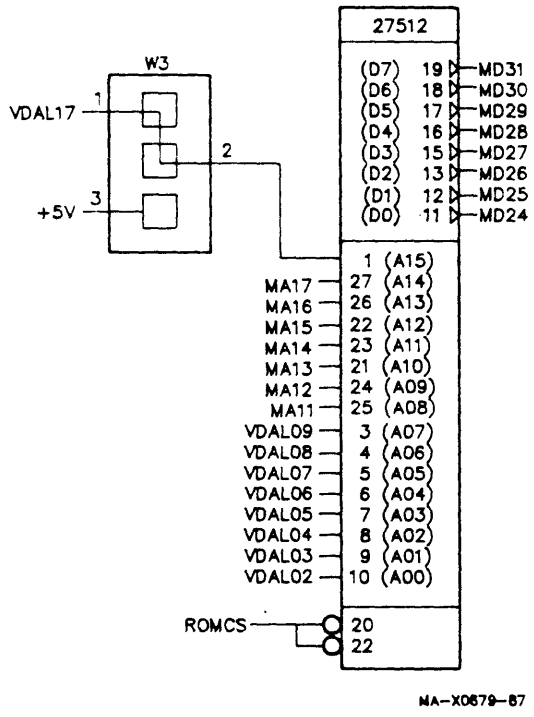
3.3.2 ROM Memory

The system module ROM contains processor restart, diagnostic and console code, and I/O device drivers. There is also a separate ROM located on the system module that contains the ThinWire Ethernet hardware address. All option modules contain their own separate ROM memory as well.

3.3.2.1 System Module ROM

The system module contains four 28-pin ROM sockets which can hold 128K bytes or 256K bytes of data depending upon the type of ROM chips used. Jumper W3 on the system module adjusts the sockets for 27256 or 27512 (or equivalent) ROM chips. When 27256 chips are used, W3 must be on pins 2 and 3. When 27512 chips are used, W3 must be on pins 1 and 2 as shown in Figure 3-27. ROM data appears at physical address 2004.0000 through 2007.FFFF (256 Kbytes). If 27256 chips are used, their image appears twice in this address space. The data path to the system module ROM is 32 bits wide.

Figure 3-27: System Module ROM Circuit Diagram (High Byte)



When the address comes out of the CPU, part of the address VDAL09:02 is latched in the 74F373 8-bit latch and the entire address is latched inside the DC524 standard cell. The standard cell decodes the ROM chip select line and puts out a partial ROM address on the MEMAD lines. This partial address combines with the latched high order address from the 74F373 at the ROM latch to form the whole ROM address. The ROM instruction is then put out onto the MD31:00 bus which is then buffered onto the BDAL bus and on into the CPU chip over the VDAL bus. The ROM is also selected during the interrupt cycle when the standard cell puts out the interrupt address lines onto the high order address lines and a partial address on the MEMAD lines to from the whole address. The interrupt vector is then put out on the MD bus for the CPU to read.

The system ROMs are word addressed (16 bits) by the CPU for the low byte address and by the DC524 standard cell for the high address byte (excluding A15). Address bit A15 is controlled by jumper W3 which allows the VDAL17 bit from the CPU to address it when 64K x 8 ROMs are used or address bit is pulled high by +5Vdc when 32K x 8 ROMs are used. During an interrupt cycle, the standard cell takes control of address lines A14:A to send the interrupt vector address to the ROM. The ROM then outputs the starting address location of the interrupt service routine for the interrupting device. The ROMs chip select and output enable control signals are controlled by the standard cell. Refer to Section 3.5.2 for information on memory timing cycles.

There are two types of information required in the system ROMs. One type is a per part, or per chip, information which contains general information about each chip such as the ROM index number and the checksum for each chip. The second type of information is from the set, or collective ROM storage, of all four chips. The main portion of the ROM which holds the software and tables is contained in the set of the ROMs. Figure 3-28 shows the format and starting addresses of the sections within the system ROM and also whether the section is used on a per part or as a set basis. Table 3-6 lists physical addresses in the ROM that have fixed uses.

Figure 3-28: System ROM Contents Layout

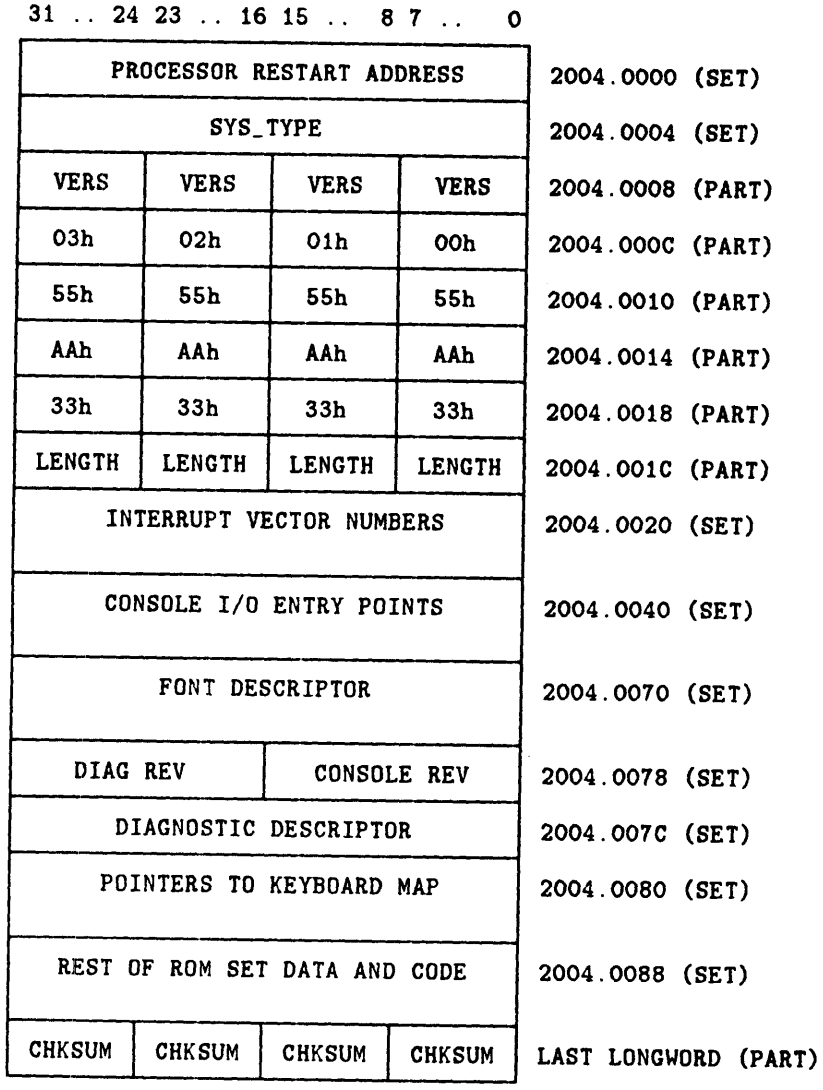


Table 3-6: Fixed ROM Address Allocations

Address	Description of Firmware
2004.0000	Processor restart address. The hardware begins execution at this address at power-up, at execution of a kernel mode halt instruction, when a break signal is received from the diagnostic console device, when the halt button is pressed, or when the CPU detects a severe corruption of its operating environment.
2004.0004	SYS_TYPE. This longword is the system type register. The value for the VAXstation 2000 and MicroVAX 2000 is 0400.0000 as described in Section 3.3.2.1.1 below.
2004.0008	Version. This field contains the low eight bits of the version number of the console code for the system firmware. The same value appears in each of the four ROM parts so that a set of chips may be verified to be compatible.
2004.000C	ROM index number. This value indicates the position of the ROM part among the set of ROMs used to implement the firmware. This value ranges from zero for the low byte through three for the high byte.
2004.0010 through 2004.0018	Manufacturing check data. These three bytes are used for a quick verification check of the ROM. The data are 55 h, AA h, and 33 h respectively.
2004.001C	ROM part length. This field indicates the length of the ROM part. It is the number of bytes in the ROM in Kbytes, for example, a 64K byte ROM has the value 64. Note that the number of bytes in the ROM set is four times this value, since there are four ROM parts in the system firmware ROM set.
2004.0020	Interrupt vector numbers. These eight longwords are used by the hardware as part of its interrupt processing. When a device generates an interrupt, the interrupt controller in the DC524 standard cell sends an interrupt vector to the system ROM so the ROM can then send the CPU the starting address of the interrupt software routines to service the device. The following list indicates the vector generated by the standard cell and the device needing servicing.

Table 3-6 (Cont.): Fixed ROM Address Allocations

Address	Description of Firmware

Vector	Interrupt source
0000.03FC	Disk controller
0000.03F8	Tape controller
0000.0248	Video controller secondary
0000.0244	Video controller end of frame
0000.0254	Network controller secondary
0000.0250	Network controller primary
0000.02C4	Serial controller transmitter
0000.02C0	Serial controller receiver

2004.0040	Console I/O routines. There are eight I/O routines provided in the system ROM. Entry points for these routines are located at longword intervals in this area.
2004.0070	Font descriptor. The system ROM contains an 8x15 character font for each graphic character in the DEC multinational character set. This font is used by the system firmware to display characters on the monochrome bitmapped display. The first longword of this descriptor is the size of the font table in bytes. The second longword is the physical address of the beginning of the fonts.
2004.0078	System console firmware revision number. This word contains the system console firmware revision number as an unsigned integer.
2004.007A	System diagnostic firmware revision number. This word contains the system diagnostic firmware revision number as an unsigned integer.
2004.007C	Diagnostic descriptor. This longword contains the physical address of the beginning of the system level diagnostic boot block. A value of zero indicates that there is no system level diagnostic present in the system ROM.

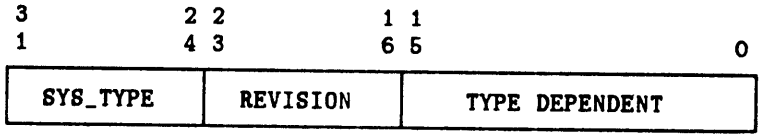
Table 3-6 (Cont.): Fixed ROM Address Allocations

Address	Description of Firmware
2004.0080	Pointers to keyboard map. These two longwords point to the tables used in translating LK201 main array keycodes to character codes. The first longword contains the physical address of the beginning of the keyboard tables. The second longword contains the physical address of the beginning of the keyboard mapping tables.
2004.0088	ROM specific data and code. This space is used for specific data needed by the system and can be updated and expanded as needed.
Last Longword	Checksum. Each ROM part contains a simple eight-bit add and rotate checksum in its last byte.

3.3.2.1.1 System Type Register (SYS_TYPE)

The SYS_TYPE register is a read-only longword in the system ROM at physical address 2004.0004. It has the format shown in Figure 3-29. The SYS_TYPE field has a value of 04h which indicates that this is a VS410 system module. The revision and type dependent fields must be zero.

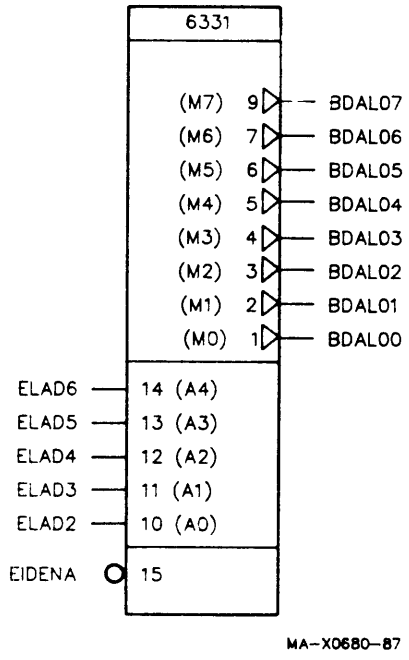
Figure 3-29: System Type Register (SYS_TYPE)



3.3.2.2 ThinWire Ethernet Address ROM

A 32-byte ROM on the system module contains a unique ThinWire Ethernet network address for the system. Data from this ROM is read in the low-order bytes of 32 consecutive longwords at physical addresses 2009.0000 through 2009.007C. The network address occupies the first six bytes (addresses 2009.0000 through 2009.0014). The byte at 2009.0000 is the first byte to be transmitted or received in an address field of an Ethernet packet. Its low-order bit (bit 0) is transmitted or received first in the serial bit stream. This ROM is installed in a socket so it can be removed from a failing system module and reinstalled on the new system module. Figure 3-30 shows the circuit diagram of the Ethernet address ROM.

Figure 3-30: ThinWire Ethernet Address ROM diagram



3.3.2.3 Option Module ROM

Each option module is required to have ROM memory that contains a standard signature to identify the option, as well as firmware initialization and diagnostic code. Four standard address ranges are defined for these ROM memories, each spanning 256K bytes. The system firmware and any operating system software that searches to determine what options are installed in a particular system should examine the signature area of each of these four ROM address ranges to see whether a valid option ROM is present and, if so, what type of option. The address ranges allocated in system module ROM are listed in Table 3-7.

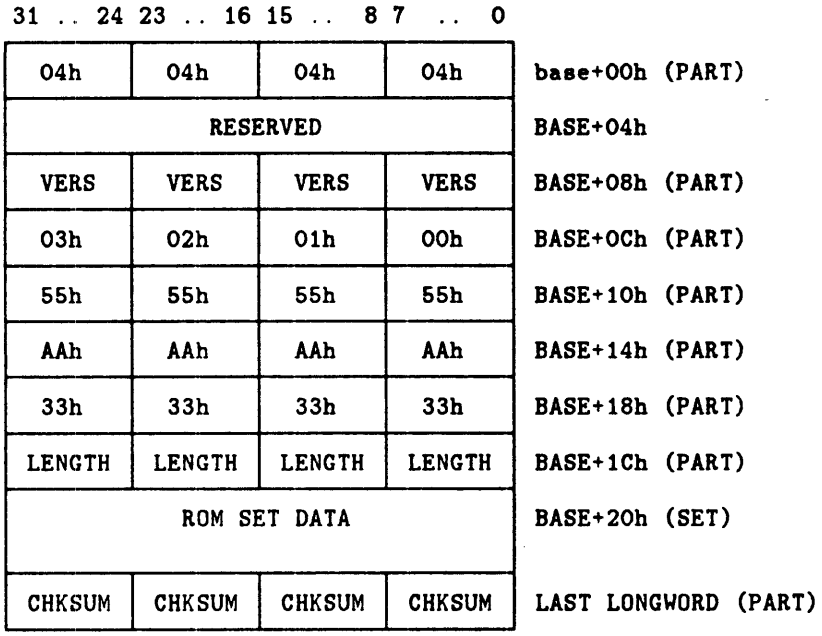
Table 3-7: ROM Address Locations Option Module ROMs

Address Range	Definition
2010.0000 to 2013.FFFF	Network option
2014.0000 to 2017.FFFF	Graphics video or serial line option
2018.0000 to 201B.FFFF	Future co-processor
201C.0000 to 201F.FFFF	Reserved

Each option module is required to have at least one ROM chip, which must be connected to the low-order byte (data lines 7:0) of the data bus. The first byte must contain the starting address of the address range. Its data is read in the low-order byte of each longword address. If there is only one ROM on the option module, then bits 31:8 of each longword are unpredictable. If two chips are used, they should be connected to data lines 15:0 of the data bus. Bits 31:16 are unpredictable. Four chips allow full use of the data bus and direct execution of code in the ROMs. Three-chip configurations are not allowed. If the size of the ROM is less than 256 Kbytes (for instance, each chip stores less than 64 Kbytes), the ROM image may repeat in the address range.

The format of the option ROMs contents, assuming there are four as shown in Figure 3-31, is setup similar to the system ROMs on the system module. The exception is the first longword that contains four bytes, each of which contains the value 04h to indicate the number of ROM chips on the option module. Another exception is the set contents of the ROM which is also described in this section.

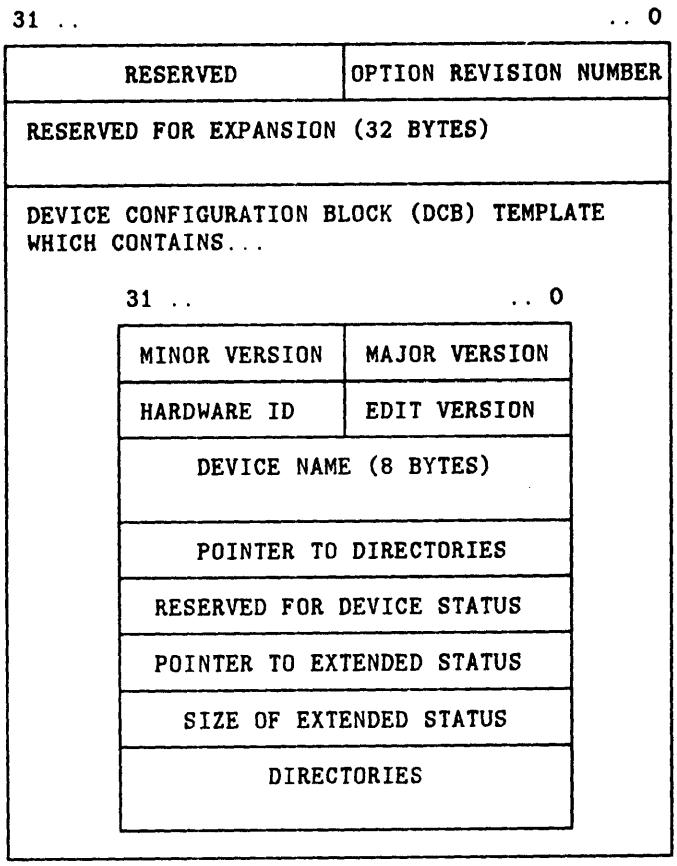
Figure 3-31: Option ROM Address Allocation



3.3.2.3.1 Option ROM Set Format

For options that use only one or two ROM chips, the data from these ROM chips must be moved into RAM. An option with four ROM chips uses the full 32-bit ROM data path and may not have to be moved. The offset to the beginning of the data in the collective set depends both on the number of ROM parts used for the option and whether the header information (ROM part data, eight bytes per chip) is included. For one chip, the header size is 08h bytes; for two chips it is 10h bytes; and for four chips it is 20h bytes. Figure 3-32 shows the set contents within the option ROM.

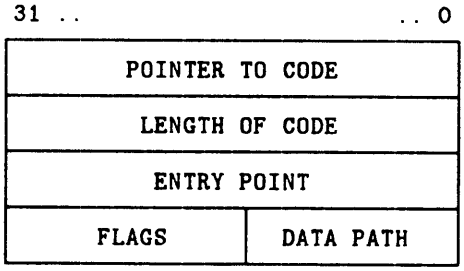
Figure 3-32: Option ROM Set Contents



Each device in the system, including optional hardware, has its own data structure called a device configuration block (DCB), which is integrated into the main configuration table (MCT) during power-up initialization. The DCB contains static and dynamic data, and pointers to code required for the device. There is a predefined set of routines used for diagnostics and console device support that must be implemented by each device. Each option must provide a template DCB for the device supported by the option. This contains information used by ROM startup code to integrate the device into the systems diagnostic structure, and information used by the next level of testing to identify the device and its capabilities.

There are six directory entries required for each option: one each for the selftest code, system exerciser code, utilities, console support, unjam, and system exerciser console support. Each directory has the format shown in Figure 3-33.

Figure 3-33: Option ROM DCB Directory Contents



3.4 Time-of-Year Clock (TOY)

The time-of-year clock (Figure 3-34) is an MC146818 CMOS watch chip that keeps the date and time of day, and contains 50 bytes of general purpose RAM. A 32.768 kHz time base oscillator provides the clock input and a rechargable nickel-cadmium battery provides power to the chip and oscillator while system power is off. The watch chip uses an LS646 transceiver to buffer and control the data and addresses to and from the CPU bus. Data from the watch chip is used to determine the date and time during the power-up of the system. See Figure 3-35.

Figure 3-34: Time-of-Year Clock

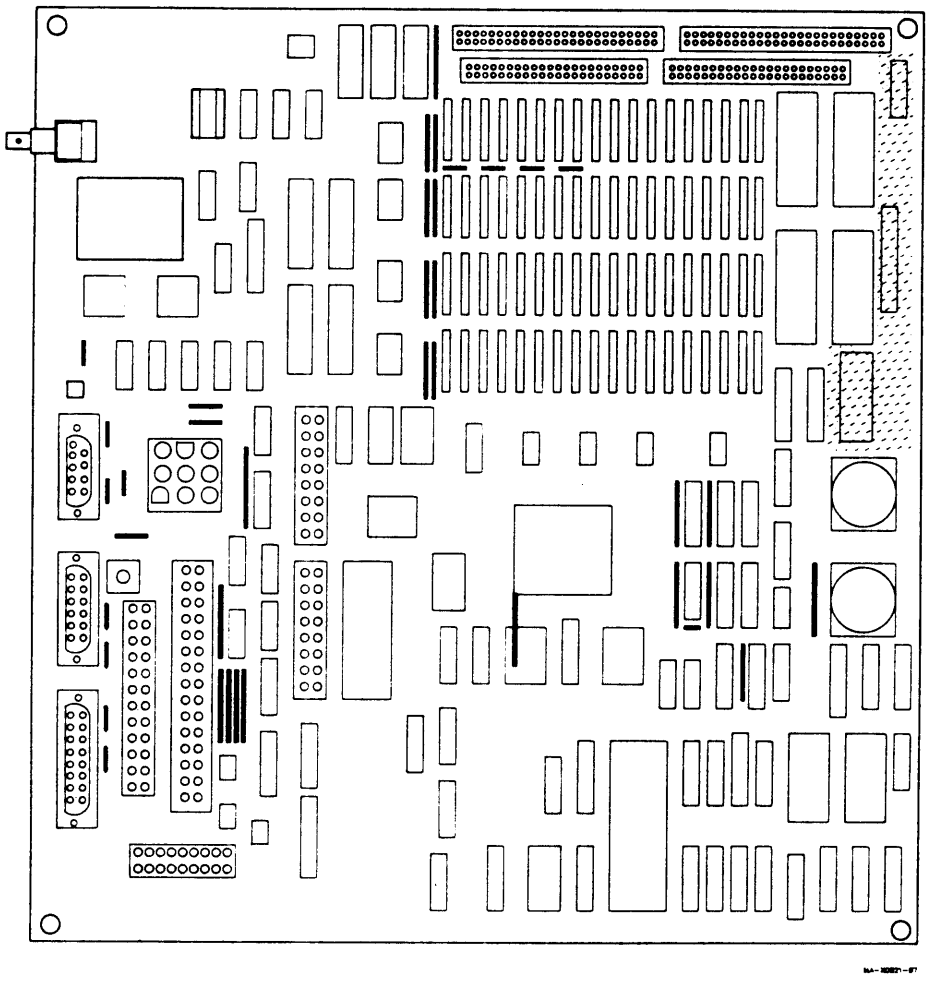
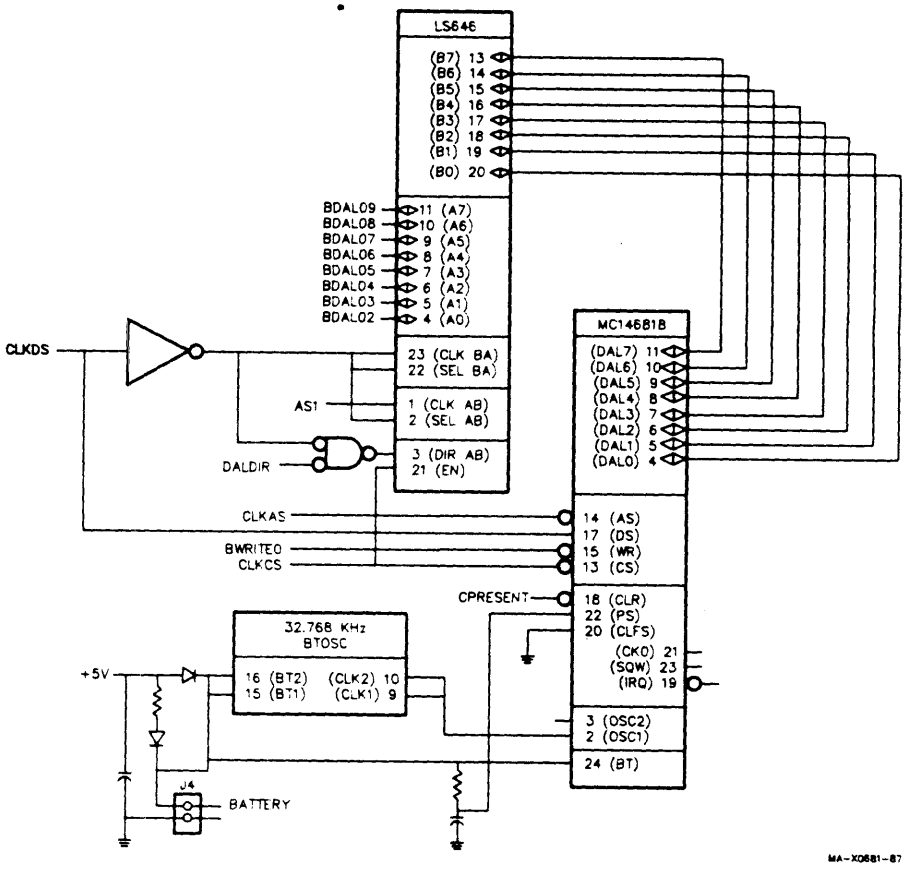


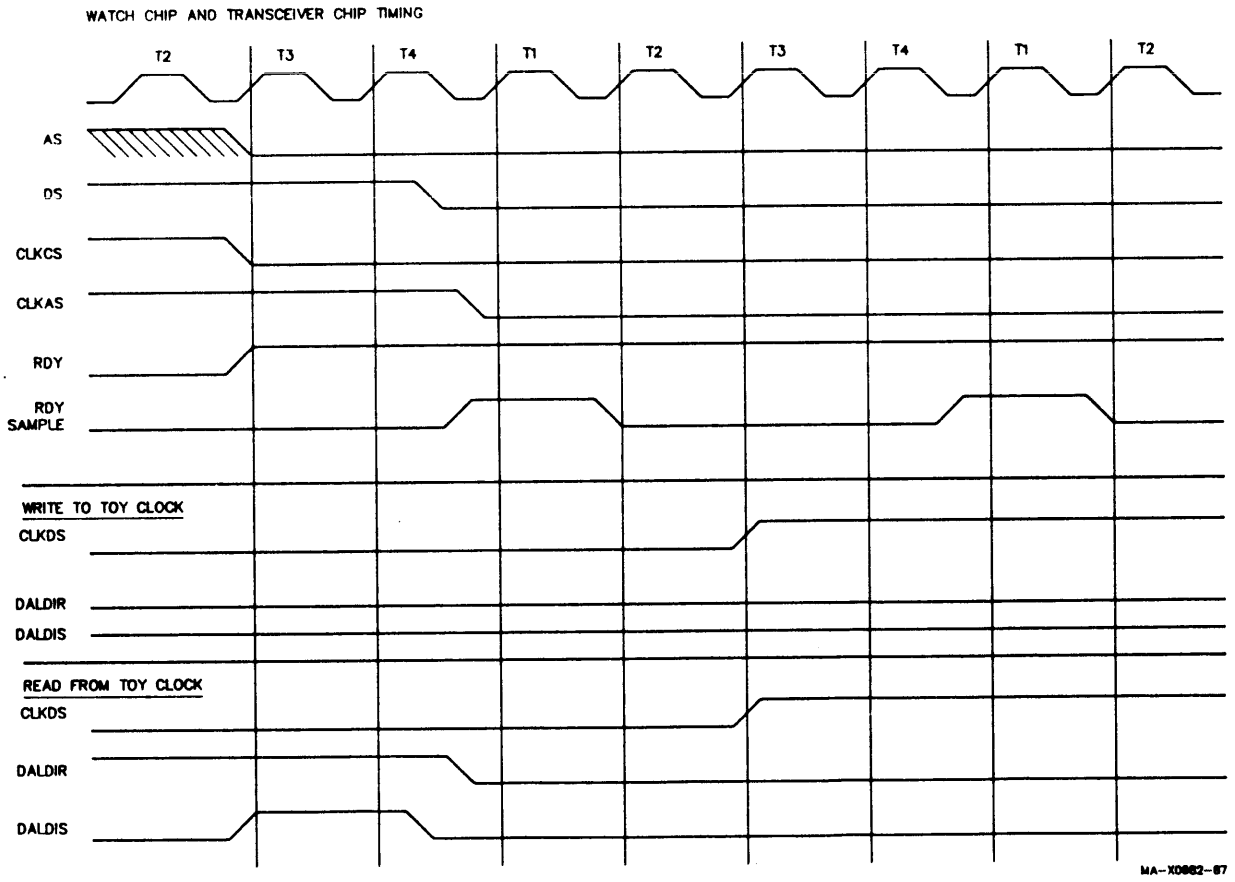
Figure 3-35: Watch Chip and Transceiver Chip Diagram



3.4.1 Watch Chip Theory of Operation

The watch chip uses an 8-bit data and address bus for reading and writing to the 64 eight-bit register storage. The first ten registers contain date and time information, the next four registers control the operation and give a status of the chip, and the last fifty registers are general purpose RAM registers used by the system firmware. This bus is controlled by four discrete signals. They are the address strobe (AS), data strobe (DS), write (WR), and chip select (CS) signals. The DC524 standard cell controls the AS, DS, and CS signals by the CLKAS, CLKDS, and CLKCS signals and the WR signal is controlled by the CPU chips BWRITE0 signal. Figure 3-36 shows the timing diagrams used to read from and write to the watch chip. The transceiver chip is used to buffer the CPU data and address bus to and from the watch chip. It is enabled by the same chip select signal to the watch chip. When enabled, the direction of data or address flow is determined by the DALDIR signal from the DC524 standard cell. The first half of a watch chip read or write cycle latches the address into the watch chip. The DALDIR signal determines whether or not the second half of the cycle is a read, asserted, or a write, unasserted, cycle. For a read cycle, the data from the watch chip is buffered directly onto the CPU bus. For a write cycle, the data from the CPU bus is buffered directly into the watch chip and stored in the latched address location.

Figure 3-36: Watch Chip and Transceiver Chip Timing



The watch chip uses a 32.768 KHz clock crystal as the time base for the time functions. Once every second, the time registers are put into update mode and are incremented by one second. The time and date registers must not be accessed by the program when in update mode. Update mode uses 1948 microseconds to complete the updates. The program must check the update in progress (UIP) bit in register WAT_CSRA to determine whether or not the chip is in update mode before attempting to access the time and date registers. To set the date and time registers the update mode must be halted by setting the set time (SET) bit in register WAT_CSRB. The SET bit, when set to 1, allows the program to set the date and time registers without being interrupted by the update mode cycle. Once the date and time registers are set, the program must reset the SET bit to 0 to start the update mode once again.

The clock crystal and the watch chip are protected from a power loss by a rechargeable nickel-cadmium battery connected to the supply voltage pin (BT). If the system loses power, or when the system is switched off, the clock crystal and the watch chip are powered by this battery so they retain the current time and date and also to retain the contents of the fifty RAM registers. If the battery voltage drops below the level needed to sustain the contents of the registers, the valid RAM and time (VRT) bit in register WAT_CSRC clears to zero to invalidate the contents of the registers. The program must check this bit during the power-up initialization to determine the validity of the contents of the watch chip registers. During power-up, the reset (CLR) signal is held low to allow the system supply voltage to stabilize. The CLR signal does not affect the clock, date, or RAM contents within the chip.

3.4.2 Watch Chip Registers

The watch chip contains 64 eight-bit registers. Ten of these contain date and time data, four are control and status registers, and the remaining 50 provide general purpose RAM storage for the system firmware. The registers occupy 64 consecutive longwords at address space listed in Table 3-8. Bits 9:2 are used in each register for data storage. Bits 31:10 and 1:0 are ignored on writing and undefined on reading.

Since each register spans two bytes on the CPU bus, only word or longword access instructions may be used to manipulate these registers. The effects of using byte access instructions are undefined. Instructions for modifying bits BBSS, BBSC, BBCC and BBCS must not be used because they generate byte-access read-modify-write cycles which corrupt the portion of the register that is not in the byte being accessed.

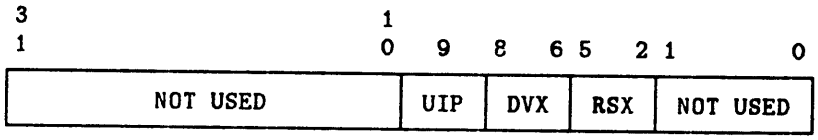
Table 3–8: Watch Chip Register Addresses

Address	Name	Definition
200B.0000	WAT_SEC	Time seconds, 0..59
200B.0004	WAT_ALMS	Alarm seconds (not used)
200B.0008	WAT_MIN	Time minutes, 0..59
200B.000C	WAT_ALMM	Alarm minutes (not used)
200B.0010	WAT_HOUR	Time hours, 0..23
200B.0014	WAT_ALMH	Alarm hours (not used)
200B.0018	WAT_DOW	Day of week, 1..7
200B.001C	WAT_DAY	Day of month, 1..31
200B.0020	WAT_MON	Month of year, 1..12
200B.0024	WAT_YEAR	Year of century, 0..99
200B.0028	WAT_CSRA	Time base divisor
200B.002C	WAT_CSRB	Date mode and format
200B.0030	WAT_CSRC	Interrupt flags (not used)
200B.0034	WAT_CSRD	Valid RAM and time flag
200B.0038		First byte of RAM data
200B.00FC		Last byte of RAM data

3.4.2.1 Control and Status Registers

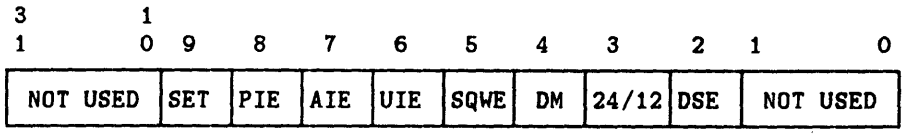
Figure 3-37 shows the format of the time base divisor (WAT_CSRA) register. Figure 3-38 shows the format of the date mode and format (WAT_CSRB) register. Figure 3-39 shows the format of the valid RAM and time (WAT_CSRD) register.

Figure 3-37: Watch Time Base Divisor (WAT_CSRA)



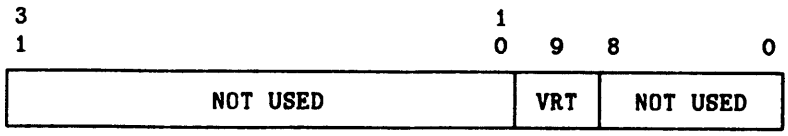
Data Bit	Definition
31:10	Not used. Ignored on writing and undefined on reading.
UIP	Update in progress (bit 9). This read-only bit indicates when the date and time registers are being updated and are hence unstable. It is set to one 244 microseconds before the beginning of an update cycle and remains one until the cycle is complete.
DVX	Time base divisor (bits 8:6). These read/write bits set the amount by which the time base oscillator input to the watch chip is divided. These bits must be set to "010" to accomodate the 32.768 KHz time base in this system.
RSX	Rate select (bits 8:6). These read/write bits select the rate at which the watch chip generates periodic interrupts. Since this feature is not used, these bits must be set to zero (0000) to disable it.
1:0	Not used. Ignored on writing and undefined on reading.

Figure 3-38: Watch Date Mode and Format (WAT_CSRB)



Data Bit	Definition
31:10	Not used. Ignored on writing and undefined on reading.
SET	Set time (bit 9). When this read/write bit is zero, the time and date registers are updated once per second. When this bit is one, any update cycle in progress is aborted and updates are inhibited so that a program can set new date and time values.
PIE	Periodic interrupt enable (bit 8). Must be set to 0.
AIE	Alarm interrupt enable (bit 7). Must be set to 0.
UIE	Update interrupt enable (bit 6). Must be set to 0.
SQWE	Square-wave enable (bit 5). Must be set to 0.
DM	Data mode (bit 4). This read/write bit selects the numeric representation in the time and date registers. If DM is one, the data format is binary. If DM is zero, the data format is two 4-bit decimal digits (BCD).
24/12	Hours format (bit 3). This read/write bit selects the format of the WAT_HOUR and WAT_ALMH registers. A value of 1 selects 24-hour mode. A value of 0 selects 12-hour AM/PM mode. In the latter case, bit 7 of the hours registers is 0 for AM and 1 for PM.
DSE	Daylight saving enable (bit 2). This read/write bit is 0 for normal operation. If set to 1, two special time updates occur: on the last Sunday in April the time increments from 01:59:59 AM to 03:00:00 AM, and on the last Sunday in October when the time first reaches 01:59:59 AM, it changes to 01:00:00 AM.
1:0	Not used. Ignored on writing and undefined on reading.

Figure 3-39: Watch Valid RAM and Time Flag



Data Bit	Definition
31:10	Not used. Ignored on writing and undefined on reading.
VRT	Valid RAM and time (bit 9). This bit indicates whether the contents of the time and RAM registers may have been corrupted by loss of power. This bit is set to 0 whenever system power is off and the backup battery voltage drops below the value required for the watch chip to function properly. This bit is set to 1 after any read of this register (the register may not be written).
8:2	Not used. Always read as 0's.
1:0	Not used. Ignored on writing and undefined on reading.

3.4.2.2 Date and Time-of-Year Registers

The time of year is kept in six registers. They are WAT_SEC, WAT_MIN, WAT_HOUR, WAT_DAY, WAT_MON, and WAT_YEAR. A seventh register, WAT_DOW, indicates the day of the week (days are numbered from 1 (Sunday) through 7). The contents of each register may either be in binary form or BCD (two 4-bit decimal digits) as selected by register WAT_CSRB bit DM.

The time value is incremented once each second. Such an update requires 1948 microseconds, during which time the date and time register contents are unstable and should not be read by a program. Register WAT_CSRA bit UIP indicates when an update is in progress. This bit is one from 244 microseconds before the beginning of an update cycle until the cycle is complete. Therefore a program should read WAT_CSRA until it finds bit UIP zero, at which time it has at least 244 microseconds to read the date and time registers. The program should inhibit interrupts while reading the registers to ensure that an interrupt does not prolong its reading beyond the 244 microsecond window.

3.4.3 Non-Volatile RAM Storage

The 50 bytes of RAM storage are used by the system firmware. Each byte actually occupies bit positions 2 through 9 of successive longwords just like the date, time, and control registers. This section lists the type of data stored in the NVR by the system firmware. There are utilities to set the boot flags, boot device, halt action, and keyboard type. These utilities are described in the *VAXstation 2000 and MicroVAX 2000 Maintenance Guide*. Table 3-9 lists the type of data stored in the NVR. All fifty registers are cleared when an NVR failure is detected during power-up.

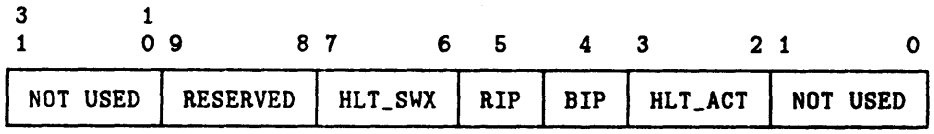
Table 3-9: Non-Volatile RAM Contents

Address	Name	Description
200B.0038	CPMBX	Console mailbox (1 byte)
200B.003C	CPFLG	Console program flags (1 byte)
200B.0040	LK201_ID	Keyboard variation (1 byte)
200B.0044	CONSOLE_ID	Console device type (1 byte)
200B.0048 through 200B.0054	SCR	Scratch RAM physical address (4 bytes)
200B.0058 through 200B.0084	TEMP	Used by system firmware (12 bytes)
200B.0088 through 200B.0094	BAT_CHK	Battery check data (4 bytes)
200B.0098 through 200B.00A4	BOOT_DEV	Default boot device (4 bytes)
200B.00A8 through 200B.00B4	BOOT_FLG	Default boot flags (4 bytes)
200B.00B8	SCR_LENGTH	Number of pages of scratch ram (1 byte)
200B.00BC	SCSI	Tape controller port data (1 byte)
200B.00C0 through 200B.00FC	Reserved	Reserved (16 bytes)

3.4.3.1 Console Mailbox Register (CPMBX)

Figure 3-40 shows the console mailbox register.

Figure 3-40: Console Mailbox Register (CPMBX)

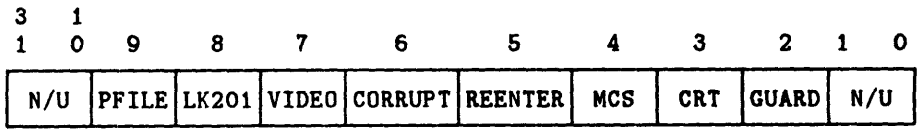


Data Bit	Definition
31:10	Not used.
9:8	Reserved for future use.
HLT_SWX	Halt switch. This is the permanent recovery action the console is to take when a processor halt occurs (except for externally generated halts such as the halt button): 0,1 - Restart. If that fails, boot. If boot fails, halt. 2 - Boot. If that fails, halt. 3 - Halt. HLT_SWX is set to 2 (Boot/Halt) when a NVR failure is detected during power-up. This field is read and written to using the console test 53 command.
RIP	Restart in progress. This restart in progress flag is set when the console attempts a restart. If it was previously set, the attempted restart is abandoned, an error message is displayed, and a boot is then attempted. This field is cleared during power-up and at entry to the console program.
BIP	Bootstrap in progress. This bootstrap in progress flag is set when the console attempts a cold restart. If it was previously set, the attempted bootstrap is abandoned, an error message is displayed, and the console program is executed. This field is cleared at power-up and at entry to the console program.
HLT_ACT	Halt action. This is the temporary recovery action the console takes when the next processor halt occurs. The action taken is the same as for the HLT_SWX field.
1:0	Not used.

3.4.3.2 Console Flags Register (CPFLG)

Figure 3-41 shows the contents of the console flags register.

Figure 3-41: Console Flags Register (CPFLG)



Data bit	Definition
31:10	Not used.
PFILE	Parameter file. This bit, when set, is used by VMB to load a parameter file along with the operating system when booting over the ThinWire Ethernet. This field is cleared when an NVR failure is detected during power-up so that no parameter file is loaded.
LK201	Keyboard type. This bit indicates whether or not the LK201 type keyboard is connected.
VIDEO	Video Flag. This bit, when set, indicates that the console display is a video display device, rather than a terminal. The particular device type is encoded in the console type register (CONSOLE_ID).
CORRUPT	Corrupted data flag. This bit is used by the console firmware during initialization.
REENTER	Reentry flag. This bit is used by the console firmware during initialization.
MCS	Multinational flag. This bit, when set, indicates that the console display understands the DEC multinational character set.
CRT	CRT flag. This bit, when set, indicates that the console display is a CRT display device.
GUARD	Guard bit. This bit is used by the console firmware during initialization.
1:0	Not used.

3.4.3.3 Keyboard Type Register (LK201_ID)

The contents of this byte is a number encoding the LK201 keyboard variant. This field is used to select the appropriate data processing keyboard map for keycode translation. This field is ignored if an attached terminal is being used as the console device. Table 3-10 lists the values available and the language that they identify.

Table 3-10: LK201 Language Values for LK201_ID Register

Value (bits 9:2)	Model Number	Language
0	LK201-xA	American
1	LK201-xB	Belgian (Flemish)
2	LK201-xC	Canadian (French)
3	LK201-xD	Danish
4	LK201-xE	British
5	LK201-xF	Finnish
6	LK201-xG	German
7	LK201-xH	Dutch
8	LK201-xI	Italian
9	LK201-xK	Swiss (French)
10	LK201-xL	Swiss (German)
11	LK201-xM	Swedish
12	LK201-xN	Norwegian
13	LK201-xP	French
14	LK201-xS	Spanish
15	LK201-xV	Portuguese

This register is set to 0 (American) if an NVR failure is detected during power-up. The console program asks the operator for the keyboard type (LK201 ID) if, at entry to the console program, the keyboard type is unknown or invalid (bit LK201 in register CPFLG is a zero or LK201_ID is out of range). This field is used only if the console device is built-in.

3.4.3.4 Console Type Register (CONSOLE_ID)

The console type register contains the type of console device as listed in Table 3-11.

Table 3-11: Console Type Register Contents

Contents	Definition of Device
0	Undefined or unknown
1	Special attached terminal on serial port 3
2	Attached terminal on serial port 0
B0	VAXstation 2000 base monochrome bitmapped display with keyboard

3.4.3.5 Scratch RAM Address Registers (SCR)

The scratch RAM address registers contain the physical address of the console program scratchpad area. This address is set during power-up by the system firmware, and should never be modified.

3.4.3.6 Temporary Storage Registers (TEMPn)

The temporary storage registers holds miscellaneous data that the system firmware needs to have stored in NVR. This temporary storage consists of twelve consecutive longwords.

3.4.3.7 Battery Check Data Registers (BAT_CHK)

The battery check data Registers are used by the system firmware as an additional check on the validity of the contents of NVR. If the battery voltage drops below the acceptable voltage level, the four battery check data registers are initialized to 55 h, AA h, 33 h, 0F h respectively during power-up initialization.

3.4.3.8 Boot Device Registers (BOOT_DEV)

The boot device registers are used by the console to store the default boot device. The device name is stored as up to four alphanumeric ASCII characters, padded to the right with 0s as necessary. If the battery voltage drops below the acceptable voltage level, these four boot device registers are initialized to all 0s during power-up initialization. These registers are read and written to using the console Test 51 command.

3.4.3.9 Boot Flags Registers (BOOT_FLG)

The boot flag registers are used by the console to store the default boot flags. If the battery voltage drops below the acceptable voltage level, these four boot flag registers are initialized to all 0s during power-up initialization. These registers are read and written to using the console Test 52 command.

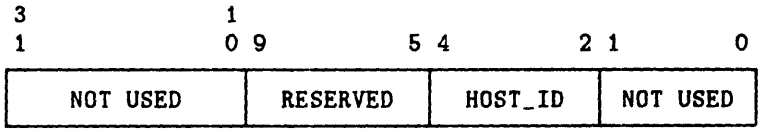
3.4.3.10 Scratch RAM Length Register (SCR_LENGTH)

The scratch RAM length register contains the number of pages of system scratch RAM. The contents of this register is determined during power-up initialization.

3.4.3.11 Tape Port Information Register (SCSI)

Figure 3-42 shows the contents of the tape port register.

Figure 3-42: Tape Port Information Register (SCSI)



Data Bit	Definition
31:10	Not used.
9:5	Reserved for future use.
HOST_ID	SCSI bus host ID address. This three bit field contains the ID address of the host on the SCSI bus. This field must always be 0 to indicate that the tape controller on the system module is the host of the bus.
1:0	Not used.

3.4.4 Initialization

When a program finds the VRT bit equal to 0, it must assume that the contents of all other registers in the watch chip are invalid. To initialize the chip, a program must do the following four steps.

1. Load register WAT_CSRB with bit SET equal to 1 to inhibit time updates and bits PIE, AIE, UIE and SQWE equal to 0 to disable unused features. Bits TM, 24/12 and DSE should be set for the desired date format.
2. Load the seven time registers with the current date and time. The addresses are listed in Table 3-8.
3. Load register WAT_CSRA to set the proper time base divisor. The DVX bits should be set to "010" and the RSX bits to "0000".
4. Load register WAT_CSRB with the same value used in step 1 except that bit SET should now be 0 to enable normal time updating.

3.4.5 Battery Backup

A nickel-cadmium battery in the system box supplies power to the watch chip and its time base oscillator while system power is off. When starting from a fully charged condition, the battery maintains valid time and RAM data in the watch chip for a minimum of 100 hours. The battery recharges while system power is on.

As long as the backup battery voltage is sufficient, the contents and operation of the watch chip are not affected by system power-on and power-off events.

3.5 DC524 Standard Cell

This section describes the operation of the DC524 standard cell Figure 3-43. Figure 3-44 shows the DC524 standard cell pinout and Table 3-12 lists the pins and their signals and describes the function of each.

Figure 3-43: Standard Cell

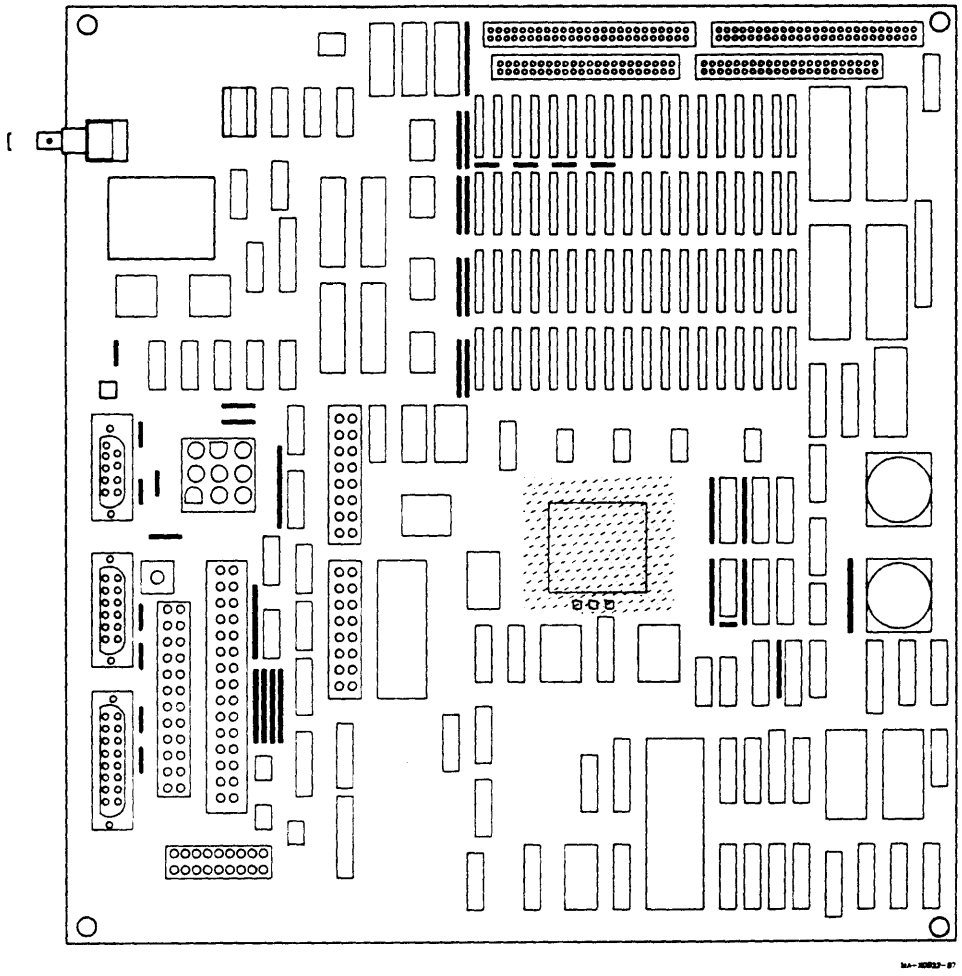


Figure 3-44: DC524 Standard Cell Pinout

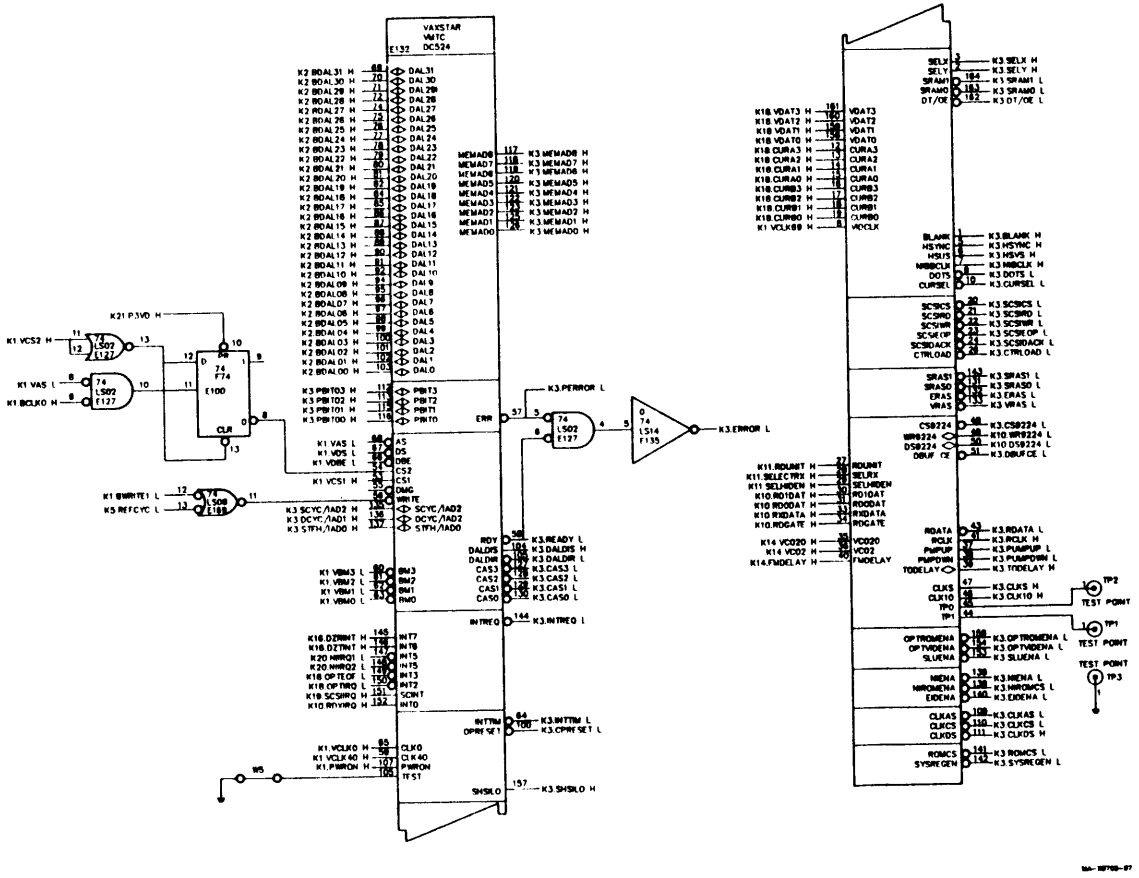


Table 3–12: DC524 Standard Cell Pinout

Pin	Signal	Description
69:72 74:82 84:92 94:103	BDAL31:00	The data and address bus (BDAL31:00) is a bidirectional time-multiplexed bus. It is connected to the CPU chip DAL31:00 bus through four 8-bit 74F245 tristate bus transceivers. These transceivers are controlled by the bus direction (DALDIR) and bus disconnect (DALDIS) signals from the standard cell.
117:123 125,126	MEMAD8:2 MEMAD1:0	These signals form the memory address bus. This bus supplies a partial address to the system ROM, system RAM, and the video RAM. See Section 3.3.2.1 for a detailed explanation of address flow.
112,113 115,116	PBIT03:02 PBIT01:00	These signals are the parity bit logic lines. They read or write parity when memory is read or written to. If a parity error occurs, the parity error signal is asserted.
57	PERROR	This signal is the parity error signal. It is asserted when a parity error is detected.
68	VAS	This signal is the address strobe from the CPU chip. It indicates when a valid address is on the BDAL bus.
67	VDS	This signal is the data strobe from the CPU chip. It indicates when valid data is on the BDAL bus.
66	VDBE	This signal is the data buffer enable signal from the CPU chip.
54	CS2	This signal is the control status 2 line from the CPU chip. This signal indicates that an interrupt cycle is in progress when this line is asserted.
53	VCS1	This signal is the control status 1 line from the CPU chip. The combination of this signal and CS2 indicate which cycle the system is in.
55	VDMG	This signal is the DMA grant line from the CPU chip. It indicates when the Ethernet network controller is in control of the system. Only the controller in the network option port has the ability for DMA.
56	WRITE	This signal indicates when a write cycle is in progress. It is asserted any time BWRITE1 is low or REFCYC is low. It is deasserted when both of these signals are high.

Table 3–12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
135 136 137	SCYC/IAD2 DCYC/IAD1 STFH/IAD0	These signals have two functions. One function is the cycle speed control function and the other is an interrupt vector address bit during the interrupt cycle. The cycle speed control funtion allows the system to access devices that are not as fast as the CPU chip. The CPU normally runs at 400 ns unless one of these lines are asserted. When the SCYC (slow cycle) line is asserted, the CPU slows down the second half of the cycle so the whole cycle runs at 600 ns. When the DCYC (double cycle) line is asserted, the CPU runs the second half of the cycle twice for a total cycle time of 800 ns. When the STFH (stall on the first half) line is asserted, the CPU is stalled on the first half of the cycle until this line is deasserted. The second function of these signals is utilized during an interrupt cycle where IAD2, IAD1, and IAD0 are controlled by the standard cell and contain the interrupt vector address of the device requesting the interrupt. This vector address is sent to the system ROM where it is decoded into the starting address of the service routine for the device requesting the interrupt.
60 61 62 63	VBM3 VBM2 VBM1 VBM0	These signals are the byte mask signals from the CPU chip. They indicate which portions of the VDAL bus are valid. Each byte mask signal validates eight lines on the VDAL bus. VBM3 validates lines 31:24, VBM2 validates lines 23:16, VBM1 validates lines 15:8, and VBM0 validates lines 7:0. Any combination of these byte mask signals may used during a cycle to validate any combination of the four 8-line segments of the VDAL bus.

Table 3-12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
58	READY	This signal is used to indicate to the CPU the end of certain cycles. The standard cell controls READY and asserts it to indicate a normal termination of the current CPU read , CPU write, or interrupt acknowledge cycles. During a CPU read cycle or interrupt acknowledge cycle, RDY L indicates that the standard cell has placed the required input data on the DAL bus. During a CPU write cycle, RDY L indicates that the information is available on the DAL bus. When the CPU chip recognizes the assertion of RDY L, it terminates the current bus cycle and proceeds. The standard cell then deasserts RDY L.
104	DALDIS	This signal controls the tri-state function of the VDAL31:00 to BDAL31:00 bus transceivers. When set high, this signal disconnects the VDAL bus (which disconnects the CPU) from the system.
105	DALDIR	This signal controls the data flow direction of the VDAL to BDAL buses. This signal allows data to flow out from the CPU chip when high and into the CPU chip when low.
127 128 129 130	CAS3 CAS2 CAS1 CAS0	These signals are the column address strobes for the memory devices in the system. The standard cell controls the memory addresses for all memory devices in the system. Each segment of the system memory is controlled by one of the CAS lines. The video RAMs are controlled by CAS1 and CAS0 lines. All four of these signals appear at the three option ports for option module memory control.
145	DZRINT	This signal indicates when the serial line controller is requesting service. This line is asserted when the serial line receiver or silo is full.
146	DZTINT	This signal indicates when the serial line controller is requesting service. This line is asserted when the serial line transmitter is done.
147 148	NIIRQ1 NIIRQ2	These signals indicate when the network controller is requesting service. The NIIRQ1 line is the primary interrupt line and the NIIRQ2 line is the secondary interrupt line for the network controller.

Table 3-12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
149 150	OPTEOF OPTIRQ	These signals indicate when the option module in the general purpose option port is requesting service. The OPTEOF is the primary interrupt line and the OPTIRQ line is the secondary interrupt line for the option module. When a color option module is installed, the primary line indicates that a video end of frame has occurred.
151	SCSIIRQ	This signal indicates when the 5380 tape controller is requesting service.
152	RDXIRQ	This signal indicates when the 9224 disk controller is requesting service.
144	INTREQ	This signal is generated by the standard cell whenever an interrupt request is received from one of the eight interrupt lines mentioned in this table.
64	INTTIM	This signal is generated by the standard cell and is the interval timer for the CPU chip. This timer provides a source of interrupts at a 10 millisecond rate.
108	CPRESET	This signal is generated by the standard cell to initialize the CPU and the system to a known state. During power-up, this signal is held low long enough for the CPU and the rest of the system to initialize, and then it is held high for normal operation.
65	VCLKO	This signal is a clock signal from the CPU chip. It is half of the VCLK40 signal.
59	VCLK40	This signal is the 40 MHz clock from a crystal oscillator.
107	PWRON	This signal is the power on line from the power-up/power-down circuitry. It is asserted high when valid power is supplied to the system.
106	TEST	This signal is the test line. It must be grounded by jumper W5 for normal operation.
157	SHSILO	This signal is used to shift the contents of the serial line silo following a read from the silo.

Table 3–12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
2 3	SELY SELX	These signals are generated by the standard cell to control the video multiplexer. These signals are decoded by the multiplexer to allow four out of sixteen video lines from the VID15:00 bus onto the VDAT3:0 bus. The data on the VDAT3:0 bus is input to the standard cell.
163 164	SRAM0 SRAM1	These signals control the video RAM. The SRAM1 line controls the high byte and the SRAM0 controls the low byte from the video RAMs.
162	DT/OE	This signal has several functions. It controls the video RAM chips for either a normal access or a video shift register update cycle. Also, it is used as a cycle type select bit for ROM cycles.
161 160 159 158	VDAT3 VDAT2 VDAT1 VDAT0	These signals are the video data bus (VDAT3:0). These four lines are from the four video multiplexers. Each video multiplexer is a four to one multiplexer which takes four signals from the video RAMs on the VID15:00 bus and outputs it to one of the VDAT lines. This type of circuit allowed the standard cell to use twelve pins for purposes other than the VDAT bus, since the VDAT bus can be multiplexed down to four lines without any timing problems.
12 13 14 15	CURA3 CURA2 CURA1 CURA0	These signals are the cursor plane A bits from the DC503 cursor sprite chip.
16 17 18 19	CURB3 CURB2 CURB1 CURB0	These signals are the cursor plane B bits from the DC503 cursor sprite chip.
8	VCLK69	This signal is the video clock input. It is from the 69 MHz video timing and refresh oscillator.
4	BLANK	This signal is generated by the standard cell and is used by the cursor chip for blanking timing information.
5	HSYNC	This signal is generated by the standard cell to synchronize the horizontal output in the cursor chip.

Table 3–12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
6	HSVS	This signal is generated by the standard cell and is used together with the DOTS signal for the video output signal to the monitor.
7	NIBCLK	This signal is generated by the standard cell and is used by the cursor chip for timing.
9	DOTS	This signal is generated by the standard cell and is used together with the HSVS signal for the video output signal to the monitor.
10	CURSEL	This signal is generated by the standard cell and is used by the cursor chip for its data strobe input signal.
20	SCSICS	This signal selects the 5380 tape controller. It is the tape controllers' chip select.
21	SCSIRD	This signal is generated by the standard cell to set up the read cycle for the 5380 tape controller.
22	SCSIWR	This signal is generated by the standard cell to set up the write cycle for the 5380 tape controller.
23	SCSIEOP	This signal is the end of process indicator. This line is asserted when the data transfer to or from the disk data buffer and the tape controller is complete. It is deasserted 150 ns after it is asserted.
24	SCSIDACK	This signal is the DMA acknowledge line to the 5380 tape controller.
25	SCSIDRQ	This signal is received from the tape controller to indicate the tape controller is requesting a DMA transfer.
26	CTRLOAD	This signal is used to load data into the DMA address register.
131 143 132 133	SRAS0 SRAS1 ERAS VRAS	These lines are the row address strobes for the memory devices in the system. The two SRAS signals are for the system RAM. SRAS0 contains the row address strobe for the first bank of RAM on the system module. SRAS1 contains the row address strobe for the second bank of RAM on the system module. The ERAS signal is the row address strobe for the expansion memory. The VRAS signal is the row address strobe for the video RAM.
48	CS9224	This signal selects the 9224 disk controller. It is the controller's chip select.

Table 3–12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
49	WR9224	This signal sets up the read cycle for the 9224 disk controller.
50	DS9224	This signal is the data strobe signal to the 9224 disk controller.
51	DBUFCE	This signal enables the disk data buffer. It is the data buffer's chip select.
27	RDUNIT	This signal selects one of the two hard disk drives when asserted.
28	SELECTRX	This signal selects the RX33 floppy disk drive when asserted and the hard disk drives when deasserted.
29	SELHIDEN	This signal selects the data rate and rotation speed of the RX33 floppy diskette drive. When asserted (high), the data rate is 500 kHz at 360 rpm. When deasserted (low), the data rate is 250 kHz at 300 rpm.
30	RD1DAT	This signal is the raw data stream from the hard disk in the expansion box. This signal is selected when SELRX is low and RDUNIT is high.
31	RD0DAT	This signal is the raw data stream from the hard disk in the system box. This signal is selected when SELRX is low and RDUNIT is also low.
33	RXDATA	This signal is the raw data stream from the floppy diskette drive in the system box.
34	RDGATE	This signal indicates that the raw data from the hard disk drives is valid.
35	VCO20	This signal is the output of the 20 MHz voltage controlled oscillator and is used for data recovery for the hard disk drives.
36	VCO2	This signal is from the output of the voltage-controlled oscillator and is used for data recovery for the floppy diskette drive.
40	FMDELAY	This signal is from the delay line that is used with the hard disk drives to provide the normal half bit delay for the phase detector.
43	RDATA	This signal is sent to the 9224 disk controller. It contains the raw read data from the disks.

Table 3–12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
41	RCLK	This signal is sent to the 9224 disk controller. It indicates when the raw read data from the disks should be examined to determine the value of the signal at that point in time.
37	PUMPUP	This signal is used to increase the frequency of the external voltage-controlled oscillator.
38	PUMPDWN	This signal is used to decrease the frequency of the external voltage-controlled oscillator.
39	TODELAY	This signal is sent to the external delay line which provides the half-bit time delay used to control the phase comparator for the hard disk drives. This output drives that delay line.
47 46	CLK5 CLK10	These signals are a 5 MHz clock and a 10 MHz clock used by the 9224 disk controller.
44 45	TP1 TP2	These signals are used to assess the performance of the phase looked loop. There is also a TP3 that is connected to ground.
156	OPTROMENA	This signal enables the ROM on the module in the general purpose option port.
154	OPTVIDENA	This signal is a general enable signal for the module in the general purpose option port.
153	SLUENA	This signal enables the four-line serial controller.
138	NIENA	This signal enables the network controller option in the DMA option port.
139	NIROMCS	This signal selects the ROM chip on the network controller option module.
140	EIDENA	This signal enables the Ethernet address ROM on the system module.
109	CLKAS	This signal is the toy-clock watch chip address strobe signal.
110	CLKCS	This signal selects the toy-clock watch chip. It is the watch chip's chip select.
111	CLKDS	This signal is the toy-clock watch chip data strobe signal.

Table 3–12 (Cont.): DC524 Standard Cell Pinout

Pin	Signal	Description
141	ROMCS	This signal selects the system ROM. It is the system ROM's chip select.
142	SYSREGEN	This signal enables the configuration and test register. This register holds the data on the configuration of the system and also the test output signal from the DC503 cursor sprite chip. This signal can also reset the disk controller and the tape controller, as well as the network controller option module and the module in the general purpose option port, without resetting the whole system.

3.5.1 Power-Up Initialization

When power is first applied to the system, the power-up/power-down circuitry holds the PWRON signal to the standard cell low as power is applied to the system and goes high after the +5 Vdc supply has risen to greater than 4.75 Vdc. PWRON is received into the standard cell by a schmitt receiver and allows the interval counter to begin counting. The standard cell then waits for 12,829 clock cycles of CLK40 (320 microseconds) to assure that all circuits are stable before deasserting the reset signal (CPRESET). This ensures that the CPU and other devices requiring initialization see an adequate number of clock cycles while in the reset state. As soon as the CPRESET signal is deasserted, the CPU chip addresses the system ROM and fetches the first instruction.

3.5.2 Memory Control

The DC524 standard cell supplies row address strobe (RAS) and column address strobe (CAS) addresses for dynamic RAMs needing either 8-bit or 9-bit addressing.

Memory cycles may be initiated by the CPU chip or the network controller operating as DMA bus master. A memory cycle begins by causing AS to change from high to low with CS2 high. All timing is then determined by the timing cycles from the input signal CLKO. Additional memory cycles are initiated to update the VAXstation 2000 video RAM memory internal shift registers and to perform memory refresh. The MicroVAX 2000 does not use the video circuits. These additional cycles are requested by counters driven from the video timing which may deassert the CPU RDY line if necessary while the memory is busy.

3.5.2.1 Multiplexed Address Signals (MEMAD8:0)

Data from several sources is multiplexed onto the memory address bus, MEMAD8:0. Data comes from the latched CPU DAL bus, the refresh address counter, the video RAM update address counter and is selected according to memory type and the current requested cycle.

3.5.2.1.1 Program RAM Cycle, CPU or DMA Read or Write

Program RAM uses 256K x 1 DRAMs as explained in Section 3.3. These chips use a 9-bit row address and a 9-bit column address. However, the only cycle to use all 9 bits is the refresh cycle. The addresses comes from the latched DAL bus as shown below.

RAS address: MEMAD8:0 = LDAL19:11
CAS address: MEMAD8:0 = LDAL10:02

3.5.2.1.2 Video RAM Cycle, CPU or DMA Read or Write

Video RAM uses four 64K x 4 video RAMs. These chips require an 8-bit RAS address taken from the latched DAL bus and two CAS addresses on the DAL bus per cycle (the memory is 16 bits wide). The high order 7 bits of the two CAS addresses also come from the DAL bus, the LSB is selected by whether the cycle is a write or a read and which half of a longword is being accessed. A longword read requires that the high word within a longword be accessed first, and a longword write requires the low word within a longword be accessed first.

RAS address: MEMADD7:0 = LDAL17:10
CAS address: (cycle1) MEMADD7:1 = LDAL09:02, MEMADD0 = 0 for WRITE(L) and 1 for WRITE(H)
CAS address: (cycle2) MEMADD7:1 = LDAL09:02, MEMADD0 = 0 for WRITE(H) and 1 for WRITE(L)

3.5.2.1.3 Video RAM Cycle, Shift Register Update

These cycles require an 8-bit RAS address which comes from an 8-bit counter that is incremented after each cycle, and a CAS address of all zeroes which indicates that the entire video shift register is to be updated. These cycles occur every 4 line times within the active video region and once immediately preceding the active video region.

RAS Address: MEMADD7:0 = VIDADD7:0
CAS Address: MEMADD7:0 = 00000000(Binary)

3.5.2.1.4 Refresh Cycles

All RAM memories in the system are updated at the same time during a RAS only refresh operation. The address comes from a 9-bit counter that is incremented after each cycle. Six cycles occur as a block during each refresh operation. These groups of six cycles occur immediately after a video RAM shift register update cycle.

RAS Address: MEMAD8:0 = REFADD8:0

3.5.2.1.5 ROM Cycles

The ROMCS signal is used to access the system ROM for both normal program operation and during interrupt acknowledge cycles when the ROM supplies the interrupt vector to the CPU. The type of cycle is indicated by the DT/OE signal. When the ROM is accessed during non-interrupt acknowledge cycles, MEMADD7:1 supply a partial address with DT/OE held high. The remainder of the ROM address is supplied from external latches. When the ROM is accessed during an interrupt acknowledge (INTACK) cycle, the number of the highest level device with an interrupt pending is output on the IAD2:0 signals and DT/OE is set to a logic 0. Having both DT/OE=0 and ROMCS active indicate that the ROM cycle is an INTACK cycle.

ROM partial address: MEMADD7:1 = LDAL16:08, and DT/OE is high

3.5.2.1.6 I/O Cycles

For cycles which perform only a single data transfer operation, the MEMADD lines are used to provide latched DAL17:10 for general peripheral controller use outside the standard cell. For I/O cycles where a double data transfer operation occurs, MEMADD0 indicates which half of a double cycle is active.

I/O partial address: MEMADD7:1 = LDAL16:10 for single I/O cycle, and MEMADD0 = cycle for double I/O cycle

3.5.2.2 Memory Control Signals

For non-DMA cycles, the standard cell bus timing starts every time AS changes from high to low with CS2 high, if neither a video RAM update cycle nor a refresh operation is in progress. For DMA cycles, where the relationship of AS generated by the DMA Bus Master to CLKO is unknown, a dual rank synchronizer is added to AS. Depending on the address which has been latched at the fall of AS, one of three row address strobes may be generated, followed by some combination of column address strobes.

All timing is generated from the CPU chip signal CLKO. The low to high transition of CLKO following the transition of AS from high to low enables any row address strobe that is required. The next CLKO low to high transition changes the address output by the cell from the row address to the column address if any row address strobe is active. For a bus read cycle, the next high to low transition of CLKO enables column address strobes, selected by the byte mask signals BM3:0. For a bus write to program RAM cycle, column address strobes are enabled at the low to high transition of CLKO following the end of the row address strobe address to allow time for the parity to be computed and output to the memory parity bits. For program RAM, the byte mask bits map into the CAS lines directly. For video RAM, where two cycles to the RAM are required for each CPU cycle, the signal CYCLE is used in conjunction with the byte masks and the WRITE signal to control CAS1:0. BM0, BM2, CYCLE and WRITE control CAS0; BM1, BM3, CYCLE and WRITE control CAS1; CAS3:2 are not used. The column address strobes are also used in some peripheral device cycles.

3.5.2.2.1 Program RAM Row Address Strobes (SRAS0 and SRAS1)

The system module has 2 megabytes of program RAM using 256K x 1 DRAM chips organized in 32-bit longwords, plus byte parity. This occupies address range 00000000:001FFFFF hex. The signal SRAS0 is generated when the CPU, or the network controller, if it is the DMA bus master, has output an address in the lower half of the address range with CS2 = 1 and when AS changes from high to low. The signal SRAS1 is generated when the address is in the upper half of the range. SRAS0/1 is active (low) from the low to high transition of CLKO following the change of state of AS from high to low until the low to high transition of DS for that bus cycle. SRAS0 and SRAS1 are both generated when a refresh cycle occurs.

3.5.2.2.2 Extended Program RAM Row Address Strobe (ERAS)

Optional program RAM can be added to the system up to a maximum of 14 megabytes using 256K x 1 DRAMS. The address range 00200000:00FFFFFF hex is decoded and the signal ERAS generated when the CPU, or the network controller operating as a DMA bus master, has output an address in range with CS2 = 1 and when AS changes from high to low. ERAS is active low from the low to high transition of CLKO following the change of state of AS from high to low until the low to high transition of DS for that bus cycle. ERAS is also generated when a refresh cycle occurs. (Note that the 14 megabyte expansion limit is the address decode limit of the standard cell and does not reflect what memory expansion modules may be available for the system).

3.5.2.2.3 Video RAM Row Address Strobe (VRAS)

The VAXstation 2000 standard video RAM supports a single screen display of 1024 x 864 pixels. This is provided by four 64K x 4 video RAM chips. The address range 30000000:3001FFFF hex is decoded and the signal VRAS generated when the CPU, or the network controller operating as a DMA bus master, has output an address in range with CS2 = 1 and when AS changes from high to low. VRAS remains active low from the low to high transition of CLKO following the change of state of AS from high to low until the low to high transition of DS for that bus cycle. As video RAM is only word wide, the memory control generates two back to back page mode memory cycles for each bus cycle, stretching the bus cycle by deasserting RDY.

3.5.2.2.4 CAS3:2 Column Address Strobes (CAS3:2)

CAS3:2 are selected by BM3:2. CAS3:2 become active (low) after SRAS, ERAS or VRAS has gone low or after an I/O device select has been asserted.

3.5.2.2.5 CAS1:0 Column Address Strobes (CAS1:0)

CAS1:0 are selected by BM3:0 during both cycle1 and cycle2 but only two of the four byte masks select CAS1:0 and this is controlled by WRITE. During cycle1, when WRITE is low, BM1:0 select CAS1:0. When WRITE is high, BM3:2 select CAS1:0. During cycle2, when WRITE is low, BM3:2 select CAS1:0. When WRITE is high, BM1:0 select CAS1:0. CAS1:0 become active (low) after VRAS or an I/O device select has been asserted.

3.5.2.2.6 DAL Bus Transceiver Direction Control (DALDIR)

The standard cell uses bidirectional bus buffers to isolate it from the CPU chip. DALDIR controls the direction of data flow through these buffers. When DALDIR is high, data is passed from the CPU chip to the system. When DALDIR is low, data is passed from the system to the CPU ship. DALDIR is high if WRITE is low or DBE is high. DALDIR is low when WRITE is high and DBE is low.

3.5.2.2.7 DAL Bus Transceiver Enable Control (DALDIS)

DALDIS enables the DAL bus buffers for data transfer operations to and from the standard cell and peripheral devices during an extended bus cycle. DALDIS is also asserted twice during bus read cycles, once while AS1 is high and DS is high to avoid possible bus contention with very fast devices being accessed and again from the time DS has gone high again until AS1 goes low to remove data from the bus as soon as possible. DALDIS is driven high when DMG is low and when a write to the system configuration register has been decoded. When DALDIS is low, the bus buffers are enabled in the direction set by DALDIR. Table 3-13 lists the functions of the DAL bus transceiver enable control and the status of the controlling signals.

An extended bus cycle is generated when the memory or peripheral device addressed is not a full 32-bits wide and it is necessary to perform two read or write operations on sequential word addresses of the device to satisfy the CPU long word data requirement. Note that longword accesses to word wide peripheral devices may cause unpredictable results.

During cycle 1 of an extended bus read operation, the least significant address of the selected device is set to a logic 1, indicating that the high 16 bits of a longword is being requested. This is done by setting MEMADD0 to a logic 1. Data is taken from the RAM or peripheral onto buffered DAL bus bits BDAL15:00 and then into the standard cell where it is stored in a temporary register, the internal data latch. During cycle 2, this stored data is output onto BDAL31:16 and the least significant address now presented to the RAM or peripheral is set to a logic 0, indicating that the low 16 bits of the long-word is now being requested. The RAM or peripheral supplies these data bits on BDAL15:00.

For a write operation, during cycle 1 the low word address of the longword pair is sent to the selected device, the CPU data on BDAL15:00 written to the selected device and the high word data on BDAL31:16 stored in the internal data latch. During cycle 2, the word address to the device is set to the high word of the longword and the data from the internal data latch placed on BDAL15:00 to be written to the device with DALDIS driven high to disable the DAL buffers.

Table 3-13: DAL Bus Transceiver Enable Control Signals

DALDIS	DMG	Cycle2	WRITE	Function
LOW	H	L	L	CPU data to the device and standard cell
HIGH	H	H	L	Standard cell data to the device
HIGH	H	L	H	Device data to the standard cell
LOW	H	H	H	Standard cell data to the CPU chip
HIGH	L	X	X	Data transfers to and from the DMA controller

3.5.2.2.8 Data Transfer/Output Enable for VRAM (DT/OE)

This output has several functions. It controls the video RAM chips to cause either a normal access (DT/OE is high as AS falls), or a video shift register update cycle (DT/OE low as AS falls). It is also used as a cycle type select bit for ROM cycles

3.5.2.3 Memory and Peripheral Timing

Memory and peripheral timing diagrams are located in Appendix A.

3.5.2.4 Control of CPU Cycle Slips

This section describes the CPU cycle timing control signals.

3.5.2.4.1 Single and Double Cycle Slips

To guarantee a single microcycle slip, the CPU chip RDY line is deasserted at the start of T3 of an extended cycle and reasserted at the end of the following T2 for a single microcycle slip, or the end of the next T2 for a two microcycle slip (TOY clock cycles).

AS is asserted (low) during state T2 following the CLKO low to high transition that starts T2. AS is re-synchronised within the standard cell using CLKO, so AS1 is set at the start of CPU state T3. The CPU samples RDY from T4 + 30 ns to T4 + 90 ns, so if RDY is de-asserted from AS1 to AS1 + 3CLKO later, a single microcycle slip will occur. In fact RDY may be deasserted up to AS1 + 5CLKO later and there will still only be a single slip.

3.5.2.4.2 Two Cycle Requests from Optional Devices

The logic used to extend the bus cycle and to generate two word cycles per extended bus cycle may be used by any optional device by its asserting the control line DCYC/IAD2. A peripheral device needing the controller to perform a double cycle must assert DCYC/IAD2 low within 100 ns of receiving its device select from the standard cell or decoding its device select separately.

3.5.3 Video Control

3.5.3.1 Video Shift Register Update and RAM Refresh

The shift registers within the four video RAM chips contain sufficient data for four display lines, thus an update of the shift registers is required every four line times (74.1 microseconds). A single video RAM cycle with DT/OE low as VRAS falls updates all shift registers. The RAS address for such an update cycle is taken from an 8-bit counter which is preset with an offset into the video RAM address space at the end of each frame and incremented after each update cycle. As there are 864 displayed lines and an update cycle occurs each 4 lines, the counter range is INITIAL_VALUE to (INITIAL_VALUE + 215). If the INITIAL_ADDRESS is greater then 40, the counter wraps to the start of the video RAM after line 864 - 4 * (40 - INITIAL_VALUE). The CAS address for update cycles is set to zero to cause all shift registers to be used.

Following the shift register update, memory refresh is performed. During each refresh operation, sufficient refresh cycles must take place to ensure that the worst case refresh interval is not exceeded. As there are a total of 216 refresh operations for each display frame time (16.67 microseconds), this requires that six RAS0-only refresh cycles must be performed during each refresh operation (256 addresses must be accessed every 4 microseconds; every 4 microseconds there are 51 shift register update cycles; 5 cycles per operation yields a worst case refresh interval of 3.9 microseconds. But there are no refresh cycles during the vertical retrace interval, hence 6 refresh cycles per operation). The addresses for these cycles come from a 9-bit binary counter, initially cleared and incremented after each refresh. (Only eight bits are used for the present RAMs, the ninth bit is for possible future RAMs using 1 megabit chips).

The sequence of video shift register update and the six refresh cycles is started by the video timing prior to the beginning of the displayed region and subsequently every four display lines until the end of the displayed region. A request for control of the memory system is posted by either of these two events. A synchronizer then monitors AS and waits for it to be high when CLKO changes from low to high. At this time a hold state is entered and DCYC/IAD1 is driven low to indicate to external logic that a video RAM update is in progress. While hold is true, any attempt by the CPU, or the network controller operating as a DMA master, to access memory (AS going low), causes the RDY line to be de-asserted and the requested memory cycle to be held off until the update/refresh cycles have been completed. If AS goes low with CS2 high while hold is true, the CPU line RDY is deasserted and a cycle counter is started. When the update/refresh cycle is completed, RDY is kept deasserted until the cycle counter indicates that the CPU is in state T2. That is, the CPU is in a state equivalent to where it asserted AS, thus the cycle continues as though it had started normally.

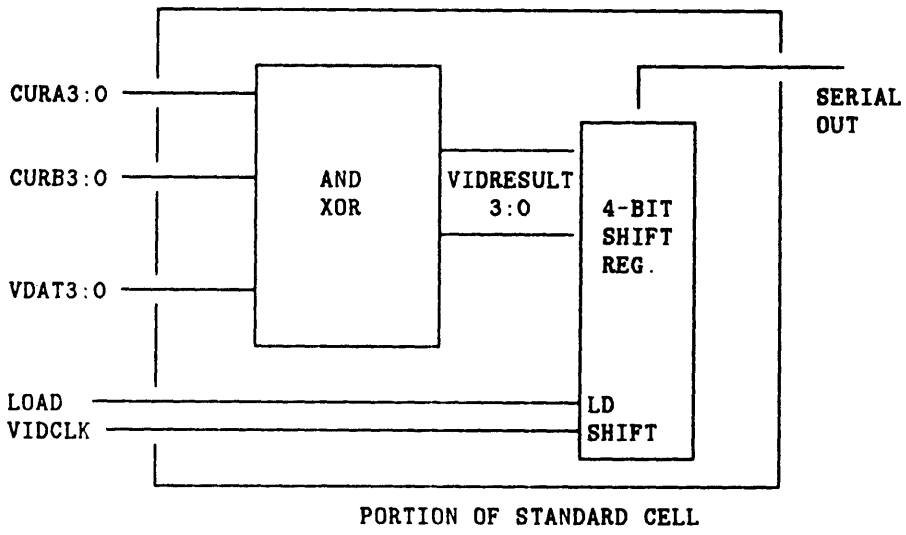
3.5.3.2 Video Timing Diagrams

The video timing diagrams are located in Appendix A.

3.5.3.3 Video RAM and Cursor Data Combination and Output

Figure 3-45 shows a simplified block diagram of the video RAM and cursor data combination and output circuits.

Figure 3-45: Video RAM and Cursor Block Diagram

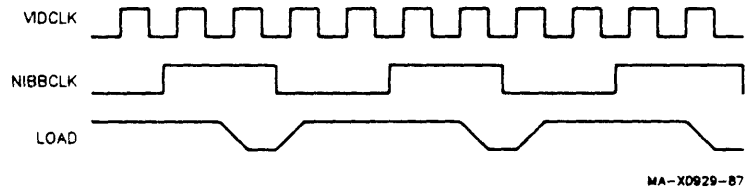


3.5.3.3.1 Video RAM Input Data (VDAT3:0)

The 16-bit output from the video RAMs are multiplexed to four bits which are input to the standard cell on VDAT3:0. VDAT0 is the first of the four bits output in the serial dot stream.

The basic video clock input to the standard cell is VIDCLK. It is divided by four to produce NIBBCLK. This is the count input to the horizontal timing generation and is also output to the cursor chip. Data input to the standard cell is four bits wide as described above. The four bits are selected from the 16 bits from the video RAMs by the two signals SELY and SELX. During one cycle of NIBBCLK, the current four bits of data are converted to serial output when LOAD ENABLE is asserted as shown in Figure 3-46.

Figure 3-46: Video Dot Cycle Timing



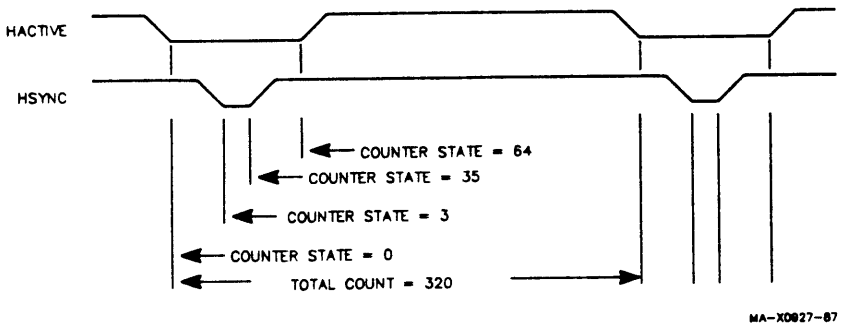
3.5.3.3.2 Cursor Data (CURA3:0 and CURB3:0)

The 8-bit output from the cursor control chip is input to the standard cell on these 8 lines and is combined with the video RAM output data to form the final video dot stream (DOTS). The combination is done on a bit basis prior to the result being loaded into a 4-bit shift register. For example, VIDRESULT0 is derived from the logical AND of VDAT0 and CURB0. The result of this logical AND is then exclusive-ORed with CURA0. This is done for all four bits.

3.5.3.3.3 Synchronization Output Pulses (HSYNC and HSVS)

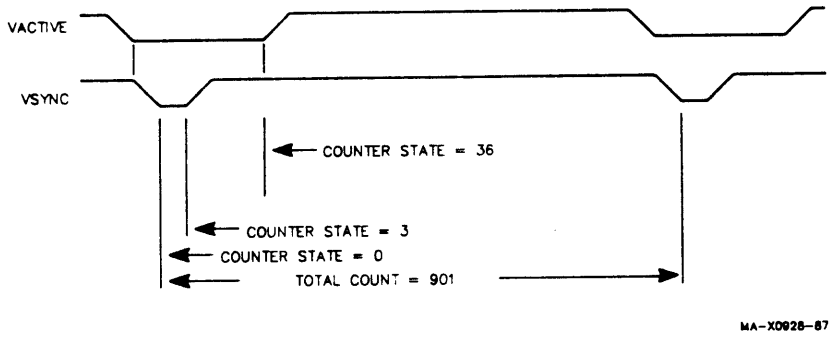
The video clock (VIDCLK) is divided by four to form NIBBCLK, this then is used as the clock to the horizontal timing which generates HACTIVE (internal signl) and HSYNC. The overflow from the horizontal timing is used as an enable to the vertical timing counters which produce VACTIVE (internal signal) and VSYNC (internal signal). HSYNC and HSVS are output to the monitor. HSVS is the logical OR of HSYNC and VSYNC. Horizontal timing is generated by a synchronous 8-bit counter whose states are decoded to generate HSYNC and HACTIVE as shown in Figure 3-47.

Figure 3-47: Horizontal Timing Generation



Vertical timing is generated by an 8-bit asynchronous counter which is clocked by the low to high transition of HACTIVE as shown in Figure 3-48.

Figure 3-48: Vertical Timing Generation



3.5.3.3.4 Video RAM Shift Pulses (SRAM1:0)

Pairs of video RAM chips are shifted separately using two shift pulses, the second of which is delayed from the first by two NIBBCLK periods (116 ns). This allows the maximum time for the video RAMs to present new data to the standard cell.

3.5.3.3.5 Shift Register Update Mode Select (DT/OE)

This output line selects whether the video RAMs are operating in normal random access read/write mode (DT/OE high when VRAS changes from high to low), or shift register update mode (DT/OE low when VRAS changes from high to low).

3.5.4 Input/Output Control

Input and output (I/O) decode is done for all the devices and for some of the options. Addresses are decoded only to the level necessary to specify the device. All I/O cycles are extended to at least 600 ns. The TOY clock cycles are extended to 800 ns and certain operations involving the Ethernet network controller cause even longer cycles to occur by controlling the SCYC/IAD2 input to the standard cell. This section describes how the I/O control signals are implemented.

3.5.4.1 Configuration and Test Register Enable (SYSREGEN)

This is an enable signal to a general purpose register that is active low when an address in the range 20020000:200200FF hex is decoded and also when CS2 goes high when AS changes from high to low. It is low from the CLKO low to high transition following the AS transition until AS returns to the high state. If WRITE is low when this address is decoded then DALDIS is driven high.

3.5.4.2 System ROM Enable (ROMCS)

The system ROM occupies I/O space and is controlled by ROMCS. ROMCS goes true (low) for addresses in the range 20040000:2007FFFF hex when CS2 is high and AS has changed from high to low when WRITE is high and DBE is low. ROMCS also goes low when DBE is low during an interrupt acknowledge cycle where CS2 is low when AS goes from high to low, so that the ROM may output a vector for the interrupting device. For these cycles, ROM address bits 2:0 are output on lines IAD2:0.

3.5.4.3 Network Option ROM Enable (NIROMENA)

To allow for the network interface controller to have on-board ROM, this output signal goes active (low) under the following conditions. An address in the range 20010000:2013FFFF hex is decoded with CS2 high, when AS has gone from high to low, and, WRITE is high and DBE is low.

3.5.4.4 Video Option ROM Enable (OPTROMENA)

A signal similar to NIROMENA, except that it is active for the address range 20140000:2017FFFF hex. It is intended for use by the option in the general purpose port.

3.5.4.5 TOY Clock Control (CLKCS, CLKAS, and CLKDS)

The time-of-year clock chip requires a longer bus cycle than other peripheral devices. Bus cycles directed to the TOY chip are extended by an additional microcycle and three control signals are generated to accommodate the slow chip timing.

- CLKCS is asserted (low) when an address in the range 200B0000: 200B00FF hex has been decoded with CS2 high, from the high to low transition of AS, to the following low to high transition of DS.
- CLKAS is asserted (low) 75 ns after CLKCS goes low and is de-asserted at the next low to high transition of DS.
- CLKDS is asserted (high) as a function of the CPUs WRITE signal. If WRITE is high (read from the TOY clock chip), CLKDS goes true, (high) 200 ns after CLKCS goes true, and remains high until the next low to high transition of DS. If WRITE is low (write to the TOY clock chip), DS goes high 200 ns after CLKCS goes low and remains high for 350 ns.

3.5.4.6 System Error, Interrupt Control and Video Control Registers

There are four registers internal to the standard cell that report and control parity error generation and checking, provide interrupt masks for all the standard peripheral devices and provide an offset into the video RAM for the start of screen.

3.5.4.6.1 Memory System Error Register (MSER)

Register MSER (address 20080004 hex) contains information relating to the parity checking of the machine. Some bits are read-only and some are read/write. Parity generation and checking is performed for all program RAM on a byte basis.

Detection of a parity error causes the CPU ERR line to be asserted (low) and held low until the end of a subsequent data stream read cycle. At the time a parity error is detected, the memory page address is latched and held until the error has been cleared. Note that there is no provision for detection of further parity errors in the interval between the time that the parity check logic has detected an error and the time that the initial error has been cleared (by a write to the MSER Register). See Section 3.3.1.5 for an explanation of this register.

3.5.4.6.2 CPU Error Address Register (CEAR)

Register CEAR (address 20080008 hex) is used to save the address at which a parity error was detected. The contents of this register are only valid when a parity error has occurred and CPU LPE is set (MSER5 = 1). This register is read-only. Bits 31:15 are read as 0 and 14:00 are the failing address bits 23:09.

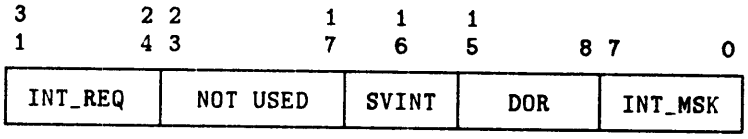
3.5.4.6.3 Diagnostic Register

The diagnostic register (address 20080000 hex) is used for diagnostic purposes. This register is a read/write and is a full 32-bits wide.

3.5.4.7 Interrupt and Video Control Register (IVCR)

This 32-bit register (address 2008000C hex) is made up of four separate 8-bit registers; the system interrupt mask register (INT_MSK), the display origin register, the single bit (the other seven bits are not used) register used to select the source of the end of frame interrupts which can be the internal video controller or the optional video controller, and the pending interrupt status register (INT_REQ). Figure 3-49 shows the contents of this register.

Figure 3-49: Interrupt and Video Control Register (IVCR)



Data Bit	Definition
INT_REQ	Read/write (bits 31:24). This 8-bit register contains the latched result of an active INT_CLR transition on any one of the eight interrupt lines. Interrupt level 7 is reported in bit 31; interrupt level 0 is reported in bit 24. A write to this 8-bit register clears any bits set according to the bit pattern in 31:24.
23:17	Not used. Read as 0.
SVINT	Read/write (bit 16). Selects the source of end of frame interrupts. If cleared, the interrupt source is the internal video controller. This bit is cleared during power-up.
DOR	Read/write (bits 15:08). These bits are loaded with a value used by the internal video controller as an offset from the base address of the video RAM. The video RAM starts at address 30000000H, and with the offset register cleared, this address maps to the first 32 pixels of the first scan line of the display. The offset register allows the first scan line data to be taken from 30000000H + DOR*400H. If the programmed value of the offset is greater than 40, the video controller wraps back to 30000000H following access to address 3001FF80H as the video RAM is only 128 kbytes. The offset register is cleared during power-up.
INT_MSK	Read/write (bits 07:00). Individual interrupt enables for the eight sources of interrupts supported by this chip. All interrupt enables are cleared during power-up.

3.5.4.8 Serial Line Controller Enable (SLUENA)

This signal is asserted (low) when an address in the range 200A0000:200A000F hex has been decoded with CS2 high when AS goes from high to low. It is used as the enable to the four line serial line controller.

3.5.4.9 Shift Silo (SHSILO)

This signal is asserted (high) for approximately 100 ns following any read to address 200A.0004. It is used externally to shift the contents of the SLU SILO following a read from the SILO.

3.5.4.10 9224 Control Signals (CS9224, DS9224, and WR9224)

These three signals control the 9224 disk controller chip.

- CS9224 is asserted (low) when an address in the range 200C0000: 200C0007 hex has been decoded and CS2 is high and AS has gone from high to low. It remains low until AS returns high.
- When DS9224 is asserted by the CPU, it is low after CS9224 has gone low, which is approximately 100 ns after DS goes low, to allow for the long address strobe setup time of the 9224 chip. It goes high again when DS goes high. When not asserted by the CPU, this pin becomes an input and may be driven by the 9224 chip to access the disk buffer RAM. When used by the tape controller chip, it again is an output, used to control data transfer to and from the tape controller chip and the disk data buffer.
- WR9224 is asserted (low) when the 9224 chip is being addressed by the CPU or when the tape control logic needs to write to the disk data buffer. It is low when CS9224 is low and WRITE is low, approximately 100 ns after DS has gone low. It returns high when DS returns high.

3.5.4.11 Tape Port Control Signals (SCSICS, SCSIRD, and SCSIWR)

Control of the tape port (SCSI) requires reading and writing registers within the tape controller chip, reading and writing the DMA address register, writing the byte count register, and specifying the transfer direction. Several address ranges are decoded as follows.

- 200C0080:200C009F is used to address read/write registers within the tape controller chip. The signal SCSICS is asserted (low) and either SCSIRD or SCSIWR is asserted depending on whether the operation is a read or write. The tape controller chip is byte wide and accepts data from or presents data to BDAL07:00.
- 200C00A0 is used to load data into the DMA address register. This register is external to the standard cell and is byte wide, write only. When this address is decoded, SCSICS is generated, followed by CTRLOAD.
- 200C00C0 is a read/write DMA byte count register. On a write, BDAL15:00 are loaded into the DMA byte count register, as selected by byte mask bits 1:0. Byte mask bits 3:2 and BDAL 31:16 are ignored. The DMA byte count register is internal to the standard cell. On a read, the contents of the DMA byte count register are returned as BDAL15:00 as selected by BM1:0.

- 200C00C4 is used to address the tape port direction bit which is a single bit write only register. SCSI DIR is loaded from BDAL00. Any reads at this address returns all zeros.

3.5.4.12 Disk RAM Buffer Control (DBUFCE)

This signal is asserted when the disk buffer RAM is addressed by either the CPU, the tape control logic or the 9224 disk controller chip. For CPU accesses, this signal goes low when an address in the range 200D0000:200D1FFF hex has been decoded and CS2 is high, and AS has gone from high to low and remains low until AS returns high. For 9224 accesses, this signal follows DS9224.

3.5.4.13 Ethernet/SID ROM Enable (EIDENA)

This signal is asserted (low) when an address in the range 20090000:2009007F hex has been decoded and CS2 is high and AS has gone from high to low and WRITE is high. It remains low until DS goes from the low to high again. It is used to access the machine ID ROM which also serves as the Ethernet address ID ROM.

3.5.4.14 Network Interface Controller Enable (NIENA)

This signal is asserted (low) when an address in the range 200E0000:200EFFFF hex has been decoded and CS2 is high and AS has gone from high to low. It becomes inactive when AS returns high. It is used as the device enable to the network controller option.

3.5.4.15 Cursor Chip Enable (CURSEL)

This signal is asserted (low) when an address in the range 200F0000:200F00FF hex has been decoded and CS2 is high and AS has gone from high to low. It becomes inactive when AS returns high. It is used as the device enable to the cursor control chip.

3.5.4.16 Video RAM Enable (SRAM0 and SRAM1)

The video RAM occupies I/O address space as described in Section 3.5.3.

3.5.4.17 Video Option Enable (OPTVIDENA)

The signal is asserted (low) when an address in the range 38000000:3FFFFFFF hex has been decoded and CS2 is high and AS has gone from high to low. It is a general enable for use by an add-on video controller.

3.5.5 Disk Control

Some of the disk control functions for both the floppy and hard (winchester) disks are implemented within the standard cell and are described here.

3.5.5.1 Floppy Disks

Transitions received from the floppy disk are synchronized and delayed using a 40 MHz clock before being presented to the 9224 for data separation to perform the nominal half-bit period delay. Signal RXDATA is synchronized to the 40 MHz clock and then delayed by 500 ns or 1 microsecond (determined by SELHIDEN) before being used to control the PUMPUP/PMPDWN control lines to the VCO circuits on the system module.

3.5.5.1.1 Density Select (SELHIDEN)

When this signal is high, it indicates that the diskette is being read or written at a data rate of 500 kHz and that the disk rotation speed is 360 rpm. When it is low, the data rate is 250 kHz and the rotation speed is 300 rpm.

3.5.5.1.2 Select Floppy Disk/Winchester Disk (SELRX)

When this signal is high, it indicates that a floppy disk is selected. When it is low, it indicates that a winchester disk is selected.

3.5.5.1.3 Read Gate (RDGATE)

This signal is from the 9224 disk controller chip and indicates that valid data is being read from the selected disk drive. It is also used to switch the phase comparator logic from the internal clock to the recovered clock.

3.5.5.1.4 Drive to External Delay Line (TODELAY)

For winchester disks, an external delay line is used to provide the nominal half-bit time delay used to control the phase comparator rather than the digital delay used with the floppy disk drives. This output drives that delay line. It is a bidirectional pin so that I/O pad driver delays on this signal and on the signal FMDELAY may be equalized.

3.5.5.1.5 Receive from External Delay Line (FMDELAY)

The output from the delay line that is used with the winchester disks is used to provide the nominal half-bit delay for the phase detector.

3.5.5.1.6 Floppy Disk Read Data (RXDATA)

Transitions received from the floppy disk are received on this signal. If SELRX is high, a low to high transition on this line causes the ARM phase detector flipflop to be set. This enables the digital half-bit time delay.

3.5.5.1.7 2 Megahertz Voltage-Controlled Oscillator (VCO2)

This signal is from the output of the VCO circuits on the system module and is used for floppy disk data recovery.

3.5.5.2 Winchester Disks

This section describes the signals associated with the winchester disk drives.

3.5.5.2.1 Drive 0 Read Data (RD0DAT)

This signal is the raw data stream from the winchester disk and is selected when line SELRX is low and line RDUNIT is low.

3.5.5.2.2 Drive 1 Read Data (RD1DAT)

This signal is the raw data stream from the winchester disk and is selected when line SELRX is low and line RDUNIT is high.

3.5.5.2.3 Select Winchester Unit (RDUNIT)

This signal selects one of two winchester disk drives as described above.

3.5.5.2.4 20 Megahertz Voltage-Controlled Oscillator (VCO20)

This signal is the output of the 20 MHz VCO and is used for winchester disk data recovery.

3.5.5.3 Common Signals

This section describes the common signals used by the standard cell.

3.5.5.3.1 40 Megahertz Clock (CLK40)

This signal is the input of a nominal square wave from an XTAL oscillator and it is divided to produce 0.5, 1, 5 and 10 MHz. All of these are used within the standard cell. The 5 and 10 MHz signals are also outputs on CLK5 and CLK10 signals, respectively.

3.5.5.3.2 Read Clock to 9224 Disk Controller (RCLK)

This signal is sent to the 9224 disk controller chip.

3.5.5.3.3 Pump UP Control Signal to VCO (PMPUP)

This signal is used to increase the frequency of the external VCO.

3.5.5.3.4 Pump Down Control Signal to VCO (PMPDWN)

This signal is used to decrease the frequency of the external VCO.

3.5.5.3.5 Read Data (RDATA)

This signal is sent to the 9224 disk controller chip.

3.5.5.3.6 10 Megahertz Clock (CLK10)

This signal is a 10 MHz clock which is generated by dividing the CLK40 signal by four. This signal is high for 50 ns and then low for 50 ns.

3.5.5.3.7 5 Megahertz Clock (CLK5)

This signal is a 5 MHz clock which is generated by dividing the CLK40 signal by eight. This signal is high for 100 ns and low for 100 ns.

3.5.5.3.8 Test Points (TP1, TP2, and TP3)

These signals are used to assess the performance of the phase locked loop. TP3 is connected to ground. TP1 and TP2 are connected to the input side of the edge-catching flip-flops of the phase comparator. The output of these flip-flops are the PUMPUP and PUMPDWN signals on the standard cell. To use them for troubleshooting the phase locked loop, set the phase locked loop to one of the reference frequencies by inhibiting the disk controller from reading data from the disks so the pulses are steady. Measure the phase error on the positive edges of the signals. It does not matter which signal (TP1 or TP2) you sync on. For the hard disks, there should be no more than 3 ns of phase error between TP1 and TP2. (Typically, it should be .5 to .2 ns.) For the floppy diskette, there should be no more than +/-14 ns of phase error.

3.5.6 Tape Control (SCSI)

The control signals SCDRQ, SCSIDACK and SCSIEOP, with the DMA byte count register (SCD_CNT) and the SCSI direction bit (SCDIR), control the operation of the tape port. When the tape controller has been programmed for a transfer to the disk buffer by the CPU chip, the transfer sequence is as follows.

1. The tape controller asserts SCSIDRQ (high).
2. The signal SCSIDACK is then generated by the standard cell. The signals SCSIRD and WR9224 are also generated at this time.
3. One CLKO period later, DS9224 is generated.
4. When the byte count register (SCD_CNT) contains FFFF, SCSIEOP is asserted.
5. DS9224 and SCSIEOP are asserted for three CLKO periods (150 ns) and then both deasserted.

6. One CLKO period later, WR9224, SCSIRD and SCSIDACK are de-asserted.

When the tape controller has been programmed for a transfer from the disk buffer by the CPU chip, the transfer sequence is as follows.

1. The tape controller asserts SCSIDRQ (high).
2. The signal SCSIDACK is then generated. The signal WR9224 is also generated at this time.
3. One CLKO period later, SCSIWR is asserted.
4. When the byte count register (SCD_CNT) contains FFFF, SCSIEOP is asserted.
5. Three CLKO periods later, SCSIWR and SCSIEOP are de-asserted.
6. One more CLKO later, SCSIDACK is deasserted.

The tape port timing diagrams are located in Appendix A.

3.5.7 Parity Generation and Checking (PBIT3:0)

For all program RAM (address range 00000000:00FFFFFF hex), parity is generated on write and checked on read as specified by the byte mask bits if parity check is enabled. Parity is not carried in the video RAM.

A parity error causes a fatal machine check. The line ERR is asserted from the time of the parity error detection until after the next data stream read cycle. ERR causes control to be passed to the ROM restart address 20040000 hex.

3.5.8 Interval Timer Interrupt Generation (INTTIM)

This signal provides a source of interrupts at a 10 millisecond rate. It counts down the 40 MHz clock.

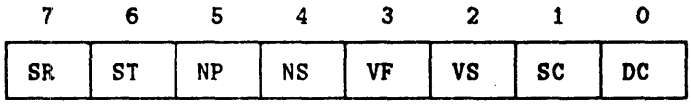
3.5.9 Interrupt Controller

The interrupt controller portion of the standard cell uses three registers to process interrupts generated by I/O devices. These registers are interrupt request (INT_REQ), interrupt mask (INT_MSK), and interrupt clear (INT_CLR). Table 3-14 lists these registers and Figure 3-50 shows the format of these registers. Note that the definition of each bit is the same in all three registers and that each bit is in the same position in all three registers.

Table 3-14: Internal Interrupt Registers

Register Name	Definition
INT_REQ	This register holds the latched interrupt requests received from I/O devices (read-only).
INT_MSK	This register contains a mask which determines which interrupt requests generate a processor interrupt (read/write).
INT_CLR	This register, which occupies the same physical register as INT_REQ, enables a program to selectively reset interrupt request bits in the INT_REQ register (write-only for INT_CLR).

Figure 3-50: Interrupt Register Formats (INT_REQ, INT_MSK, INT_CLR)



Data Bit	Definition
SR	Serial line receiver or silo full
ST	Serial line transmitter done
NP	Network controller primary
NS	Network controller secondary
VF	Video end of frame
VS	Video secondary
SC	SCSI controller
DC	Disk controller

3.5.9.1 Interrupt Request Register (INT_REQ)

The interrupt request register is an 8-bit read-only register at physical address 2008.000F. Each bit reflects the state of the interrupt request latch for one interrupt source. Bits 7:0 correspond to interrupt ranking as described in Section 3.5.9.5.

A bit in the INT_REQ register is set only by an active transition on the corresponding device's interrupt request line. The bit is set by an active transition regardless of the state of the corresponding bit in the interrupt mask register INT_MSK. However, an interrupt request is sent to the CPU only when the corresponding bits in both INT_REQ and INT_MSK are set.

A bit in the INT_REQ register is cleared by writing to the INT_CLR register with a one in the corresponding bit position. However, the highest bit set in INT_REQ is cleared automatically during a CPU interrupt acknowledge cycle as long as the corresponding bit in INT_MSK is also set. Note that INT_CLR and INT_REQ are the same physical register and that the clearing function occurs during writes to this register. Also, INT_REQ may be read at any time. Reading it does not alter the state of the system in any way. The INT_REQ is cleared to 0 during the power-up sequence.

3.5.9.2 Interrupt Mask Register (INT_MSK)

The interrupt mask register is an 8-bit read/write register at physical address 2008.000C. Each bit is a mask for one interrupt source. Bits 7:0 correspond to interrupt numbers 7:0 as listed in Section 3.5.9.5. Each mask bit is logically ended with the corresponding bit of the INT_REQ register and a nonzero result is needed before starting the priority encoder or sending the CPU interrupt request signal. If a mask bit is 0, the corresponding device's latched request (if any) is not presented to the CPU.

A 0 in a mask register bit does not prevent the corresponding device from setting its interrupt request register bit. If a request bit is set whose corresponding mask bit is 0, a CPU interrupt is not requested until the mask bit is subsequently set to 1 (assuming that the request bit has not meanwhile been cleared by writing to INT_CLR). A program which is changing from polled to interrupt servicing of a device should be sure to clear the device's bit in INT_REQ prior to setting its corresponding bit in INT_MSK in order to avoid a possible false interrupt signal to the CPU.

The interrupt mask register is cleared to 0 during the power-up sequence.

3.5.9.3 Interrupt Clear Register (INT_CLR)

The interrupt clear register is an 8-bit write-only register at physical address 2008.000F, which is used to selectively clear bits in the INT_REQ.

For each bit of INT_CLR that is a one, the corresponding bit of INT_REQ is cleared. The effect of writing to INT_CLR is transient. Its contents are not stored and writing to it does not prevent any INT_REQ bits from being set in the future.

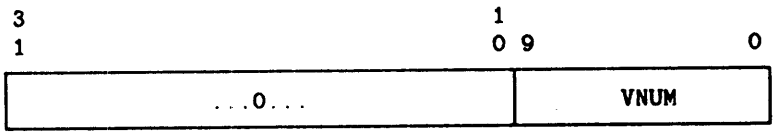
3.5.9.4 Interrupt Vector Generation

Once an interrupt is declared valid, the controller asserts the interrupt request line to the CPU. When the CPU acknowledges the interrupt, the interrupt controller sends the address of the interrupt vector to the system ROM over the address bus. This address is calculated using the interrupt number (7 through 0) of the I/O device, which also corresponds to the bit position in INT_REQ, in the following formula.

ROM address = 2004.0020 + (interrupt number * 4)

This address, once it is calculated, points to one of eight longwords in the system ROM, which holds the interrupt vector for that particular I/O device. Figure 3-51 shows the format of an interrupt vector longword in ROM.

Figure 3-51: Interrupt Vector Longword



Data Bit	Definition
31:10	Ignored. Should be 0.
VNUM	Interrupt vector number:

	Vector	Source
	2C0	Serial line controller receiver done or silo full.
	2C4	Serial line controller transmitter done.
	250	Network controller primary (network option module).

Vector	Source
254	Network controller secondary (network option module).
244	Video end-of-frame (system module or video option module, according to the VDC_SEL register).
248	Video controller secondary (video option module).
3F8	Tape controller (5380).
3FC	Disk controller (9224).
	Bit 0 is the priority level flag which selects the IPL in the CPU. If this bit is 0, the IPL is 14h. If it is 1, the IPL is 17h.

3.5.9.5 Interrupt Sources and Ranking

Table 3-15 lists the interrupt sources from highest to lowest priority. The interrupt numbers 7:0 indicate their bit positions in the INT_REQ, INT_MSK, and INT_CLR registers and also indicate their relative priority when more than one request is pending. Interrupt 7 represents the highest priority. The edge column indicates the signal transition, positive or negative, that sets the device's bit in the INT_REQ register (the opposite transition has no effect).

Interrupts 0, 1, 6 and 7 are dedicated to devices on the system module. Interrupts 2, 4, and 5 come from devices attached to option module connectors. Interrupt 3 comes from either the system module or from the video option connector, according to the setting of the VDC_SEL register.

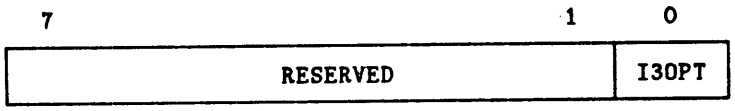
Table 3-15: Interrupt Priority Ranking

Priority	Name	Edge	Interrupt source
7	SR	Positive	Serial line controller receiver done or silo full
6	ST	Positive	Serial line controller transmitter done
5	NP	Negative	Network controller primary (network option module)
4	NS	Negative	Network controller secondary (network option module)
3	VF	Negative	Monochrome video end-of-frame or video option module according to the VDC_SEL register
2	VS	Negative	Video controller secondary (video option module)
1	SC	Positive	Tape controller
0	DC	Positive	Disk controller

3.5.9.6 Video Interrupt Select Register (VDC_SEL)

The source of the video end-of-frame interrupt signal (priority 3 in Table 3-15) is determined by the VDC_SEL register, which is a one-byte read/write register at address 2008.000E. Figure 3-52 shows the video interrupt select register.

Figure 3-52: Video Interrupt Select Register (VDC_SEL)



Data Bit	Definition
7:1	Reserved. Returns unpredictable data when read. Must be written as 0's.
I3OPT	Interrupt 3 source. If this bit is 0, interrupt 3 comes from the monochrome controller on the system module. If bit 0 is 1, the interrupt comes from the controller on the video option module. This bit is cleared to 0 during power-up initialization.

3.5.10 Monochrome Video Display Controller

The video display controller generates a monochrome image which is 1024 pixels wide by 864 pixels high. The controller consists of one bit-mapped display data plane. It can superimpose a cursor at any position on the display independently of the contents of the data plane.

3.5.10.1 Video Timing

All video timing is derived from the pixel clock crystal whose frequency is 69.1968 MHz, which yields a pixel time of approximately 14.5 ns. The timing of the synchronization and blanking signals cannot be changed by a program. Table 3-16 shows monochrome video timing.

Table 3-16: Monochrome Video Timing

Frequency Type	Frequency
Pixel	69.1968 MHz
Horizontal	54.06 kHz
Vertical	60.0 Hz

Horizontal Timing	Microseconds	Pixels
Entire line	18.50	1280
Visible raster	14.80	1024
Active line time	14.798	-
Blanking	3.70	256
Sync front porch	0.173	12
Sync pulse width	1.85	128
Sync back porch	1.676	116

Vertical Timing	Milliseconds	Lines
Entire frame	16.667	901
Visible raster	15.982	864
Blanking	0.684	37
Sync front porch	0.000	0
Sync pulse width	0.055	3
Sync back porch	0.629	34

3.5.10.2 End-of-Frame Interrupt

An interrupt request is generated at the trailing edge of each vertical sync pulse, which is three horizontal scan times after the beginning of each vertical blanking interval. (The interrupt vector is listed in Section 3.5.9.4). The time between this interrupt and the end of the vertical blanking interval is approximately 620 microseconds (34 line times). Interrupts occur at the frame rate of 60 Hz. Interrupts may be masked by clearing bit VF of the interrupt mask register (INT_MSK) to zero (See Section 3.5.9.3). Upon power-up, this mask bit is cleared to zero. In order for this end-of-frame signal to be recognized as an interrupt, the VDC_SEL register (Section 3.5.9.6) must be set to select this source rather than the video option module.

3.5.10.3 Data Plane Storage

The display data plane is stored in a 128K byte block of dual port RAM. It occupies the physical address range 3000.0000 through 3001.FFFF. Access to the RAM can be byte, word, or longword.

One displayed line of 1024 pixels is represented by 32 consecutive longwords, beginning at an address whose low-order 7 bits are all 0 (that is, a multiple of 128 decimal). Each longword appears as 32 consecutive pixels on a display line. Bit 0 of a longword (least significant) is displayed as the leftmost pixel and bit 31 (most significant) is displayed as the rightmost pixel of the 32-pixel group. Longword addresses increase from left to right across a displayed line and exactly 32 longwords are required for each line. The 128K byte data plane storage holds 1024 line images, 864 of which are visible on the display at any one time.

NOTE: *An error in the standard cell allows part of the 865th line to be visible. To fix this problem, ensure that the 32 longwords following the last display scan line contain 0.*

3.5.10.4 Display Origin Register (VDC_ORG)

The address in the data plane storage which corresponds to the top line of the display raster is determined by the 8-bit read/write register VDC_ORG, whose address is 2008.000D. This register supplies bits 16:9 of the address of the top line. Thus, the address of the first longword in the topmost displayed line is:

Address = 3000.0000 + (VDC_ORG * 512)

The visible display can begin on any 4-line boundary and wraps from the last line in the data plane storage (beginning at 3001.FF80) to the first line (beginning at 3000.0000). The contents of VDC_ORG are used at the beginning of the vertical blanking interval to reset the video controller address counter. Register VDC_ORG can be written to at any time. The contents of VDC_ORG are cleared to 0 at power-up.

Changing VDC_ORG does not affect the displayed position of the cursor sprite on the screen. The sprite's position registers operate relative to the first line displayed, regardless of what memory address it comes from.

3.5.11 Test Mode (TEST)

This signal is a general test input which modifies some internal connections to facilitate the standard cell's chip test as explained below. The standard cell is in test mode when jumper W5 is removed.

3.5.11.1 Interval Counter

This consists of four sections: divide by 10, divide by 25, divide by 25 and divide by 32 counters. These are normally cascaded to count down the 20 MHz clock to 100 Hz. In test mode, each counter has the input clock gated directly to its count input and each section output may be observed at the INTTIM output which is selected by DAL31:30 as shown in Table 3-17.

Table 3-17: Standard Cell Test Mode Addressing

DAL31	DAL30	INTTIM
0	0	Divide by 32
0	1	Second divide by 25
1	0	First divide by 25
1	1	Divide by 10

3.5.11.2 Vertical Timing

When TEST is high, the input to the vertical timing counters is changed from HACTIVE to NIBBCLK to allow a faster test.

3.5.11.3 Video RAM Shift Register Update/Refresh

When TEST is high, the count inputs to the update address counter and the refresh counter can be accessed using CLK40 and a combination of DAL02:00.

3.6 DC503 Cursor Sprite Chip

This section describes the DC503 cursor sprite chip (Figure 3-53).

3.6.1 Overview

The DC503 cursor sprite chip generates a cursor display on the video monitor. The cursor is generated from a two plane memory array within the cursor chip. Refer to Section 3.5.3 for video timing and control information. This chip is not implemented when the system module jumper is set for MicroVAX 2000 usage. Figure 3-54 shows the pinout of the DC503 cursor chip and Table 3-18 lists the chip signals and their description.

Figure 3–53: DC503 Cursor Sprite Chip

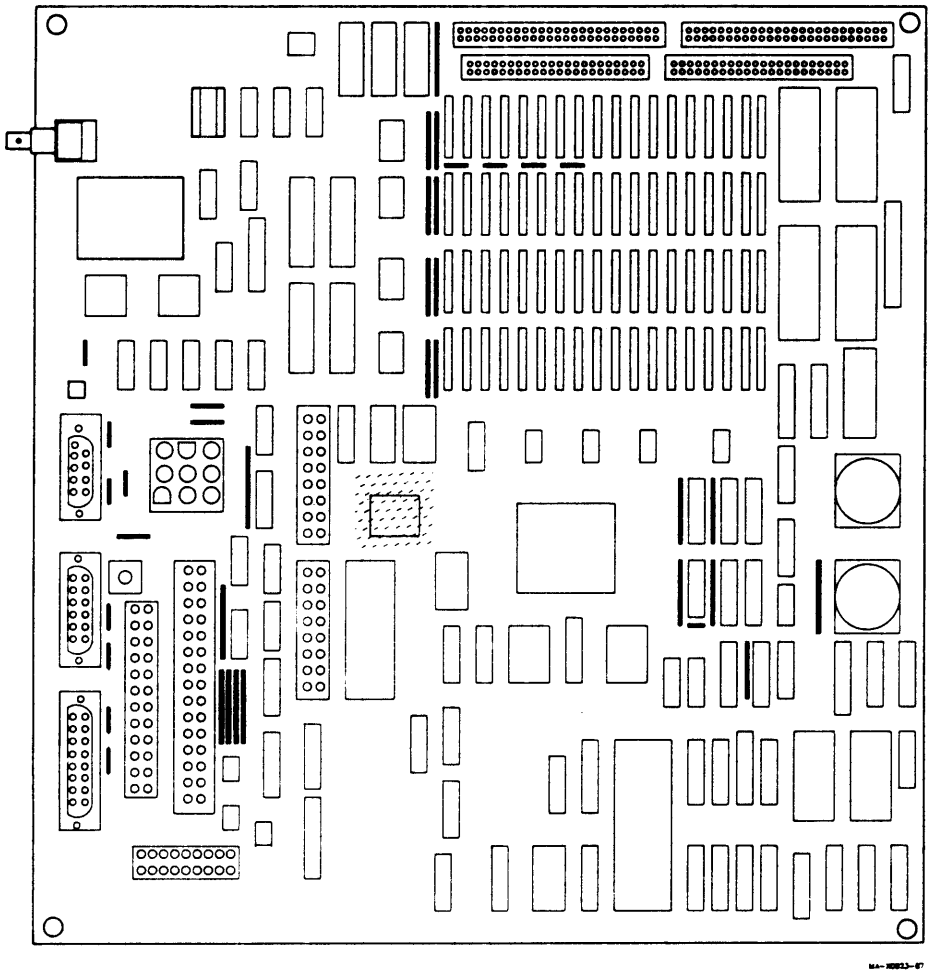


Figure 3-54: DC503 Cursor Sprite Chip Pinout

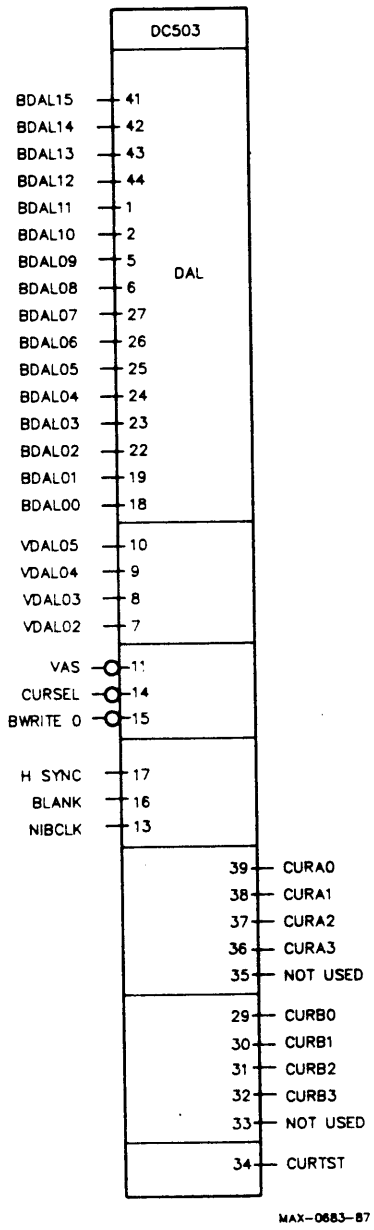


Table 3–18: DC503 Cursor Sprite Chip Pin Description

Signal	Pin	Description	Signal	Pin	Description
BDAL15	41	Data bus bit 15	BDAL14	42	Data bus bit 14
BDAL13	43	Data bus bit 13	BDAL12	44	Data bus bit 12
BDAL11	1	Data bus bit 11	BDAL10	2	Data bus bit 10
BDAL09	5	Data bus bit 9	BDAL08	6	Data bus bit 8
BDAL07	27	Data bus bit 7	BDAL06	26	Data bus bit 6
BDAL05	25	Data bus bit 5	BDAL04	24	Data bus bit 4
BDAL03	23	Data bus bit 3	BDAL02	22	Data bus bit 2
BDAL01	19	Data bus bit 1	BDAL00	18	Data bus bit 0
VDAL05	10	Address bus bit 3	VDAL04	9	Address bus bit 2
VDAL03	8	Address bus bit 1	VDAL02	7	Address bus bit 0
VAS	11	Address strobe	CURSEL	14	Data strobe
BWRITE0	15	Write enable	HSYNC	17	Horizontal sync
BLANK	16	Blank	NIBCLK	13	Clock input
CURA0	39	Plane A bit 0	CURA1	38	Plane A bit 1
CURA2	37	Plane A bit 2	CURA3	36	Plane A bit 3
CURA0	29	Plane B bit 0	CURA1	30	Plane B bit 1
CURA2	31	Plane B bit 2	CURA3	32	Plane B bit 3
CURTST	34	Test pin			

3.6.2 Cursor Coordinate Offsets

The visible raster is 1024 pixels wide in the X direction and 864 lines high in the Y direction. The nominal range of cursor coordinates is 0 through 1023 (left to right) and 0 through 863 (top to bottom). An offset must be added to nominal raster coordinate values before loading the values into the cursor position and region limits registers, because the X and Y position counters are reset at some time prior to the beginning of the visible display. The offset values are listed in Table 3–19.

Table 3-19: Cursor Coordinate Offsets

Offset	Value
X offset	216 pixels
Y offset	33 lines

For example, to display a sprite cursor with its upper left corner in pixel 100, line 300, a program must load CUR_XPOS with (100 + 216) and CUR_YPOS with (300 + 33).

3.6.3 Cursor Generation

The cursor can take two forms: a 16-bit by 16-bit pattern (sprite), or a crosshair whose lines may extend to the edges of the visible raster or may be clipped to a programmed region. The cursor hardware uses a DC 503 programmable sprite cursor chip which generates two display planes called the A and B planes. Bits from these planes are combined with bits from the data plane and the possible combinations are listed in Table 3-20.

Table 3-20: Cursor Generation Values

Data	A plane	B plane	Displayed	Cursor appearance
0	0	0	Black	Invisible
0	0	1	Black	Black
0	1	0	White	Inverted data
0	1	1	White	White
1	0	0	White	Invisible
1	0	1	Black	Black
1	1	0	Black	Inverted data
1	1	1	White	White

3.6.4 Cursor Control Registers

The cursor chip contains the following programmable elements:

- Two 16-word arrays to store a 16-bit by 16-bit sprite pattern for each cursor plane.
- X and Y position registers to control where the cursor pattern is displayed in the raster.

- Two region detectors, each of which defines a rectangle in the raster which can be used to clip the display of a crosshair cursor.
- A control register which determines how the cursor is generated.

To a program, the cursor chip appears as 12 write-only registers, each one word (16 bits) wide. These registers should always be written with word-access instructions; they cannot be read (hence read-modify-write instructions such as BIS cannot be used). The register's contents after power-up are indeterminate. The addresses and names of the registers are listed in Table 3-21.

Table 3-21: Monochrome Cursor Control Registers

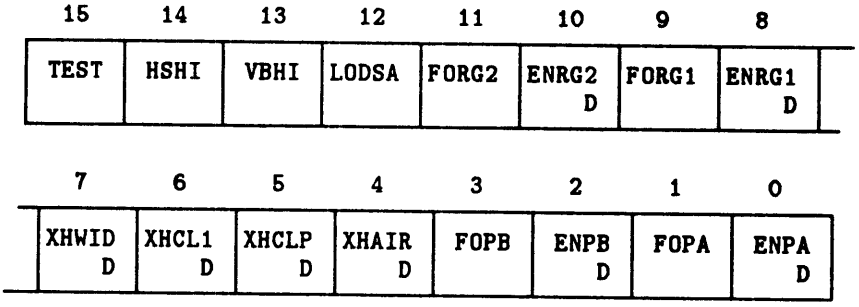
Address	Name	Note	Function
200F.0000	CUR_CMD		Cursor command register
200F.0004	CUR_XPOS	D	Cursor X position
200F.0008	CUR_YPOS	D	Cursor Y position
200F.000C	CUR_XMIN_1	D	Region 1 left edge
200F.0010	CUR_XMAX_1	D	Region 1 right edge
200F.0014	CUR_YMIN_1	D	Region 1 top edge
200F.0018	CUR_YMAX_1	D	Region 1 bottom edge
200F.002C	CUR_XMIN_2	D	Region 2 left edge
200F.0030	CUR_XMAX_2	D	Region 2 right edge
200F.0034	CUR_YMIN_2	D	Region 2 top edge
200F.0038	CUR_YMAX_2	D	Region 2 bottom edge
200F.003C	CUR_LOAD		Cursor sprite pattern load

In order to prevent unsightly effects on the display, the registers marked "D" in the Note column are buffered, as are some of the bits in the cursor command register. The processor may write into such a register or bit at any time (except within three horizontal scan times following the beginning of vertical blanking), but the new value takes effect only at the beginning of the next vertical blanking interval. Since the processor receives its end-of-frame interrupt signal three line times after vertical blanking begins, a program may ensure that it has ample time to perform a multi-register update by waiting for the end-of-frame interrupt before starting to load new values. From the time of the interrupt, it has nearly an entire frame time (16.612 milliseconds) to load the registers.

3.6.5 Cursor Command Register (DUR_CMD)

The cursor command register is a 16-bit write-only register at address 200F.0000. As in the preceding list of cursor registers, the bits marked with "D" in Figure 3-55 are buffered and do not take effect until the beginning of the next vertical blanking interval.

Figure 3-55: Cursor Command Register (CUR_CMD)



Data Bit	Definition
TEST	Diagnostic test (bit 15). This bit must be 1 for normal operation. When this bit is 0, the chip is placed in test mode, which is discussed below.
HSHI	Horizontal sync polarity (bit 14). This bit must be 1 to indicate to the chip that the horizontal sync input from the video controller is active high.
VBHI	Vertical blanking polarity (bit 13). This bit must be 1 to indicate to the chip that the vertical blanking input from the video controller is active high.
LODSA	Load/display sprite array (bit 12). When this bit is 0, the cursor sprite is displayed normally from the contents of the sprite arrays. When this bit is 1, display of the sprite is inhibited and the sprite arrays can be loaded by successive writes to the CUR_LOAD register. Upon the transition of LODSA from 1 to 0, the internal array address counter is reset so that the next write to CUR_LOAD will load the top row of sprite plane A.

Data Bit	Definition
FORG2	Force region detector 2 output to 1 (bit 11). When this bit is 1, the output of region detector 2 is forced to 1 (true). When this bit is 0, the detector operates normally.
ENRG2	Enable region detector 2 (bit 10). When this bit is 0, the output of region detector 2 is inhibited; it is 0 (false) unless the FORG2 bit is also set, which takes precedence and forces the output to 1 (true). When ENRG2 is 1, the detector operates normally.
FORG1	Force region detector 1 output to 1 (bit 09). When this bit is 1, the output of region detector 1 is forced to 1 (true). When this bit is 0, the detector operates normally.
ENRG1	Enable region detector 1 (bit 08). When this bit is 0, (false) the output of region detector 1 is inhibited; it is 0 unless the FORG1 bit is also set, which takes precedence and forces the output to 1 (true). When ENRG1 is 1, the detector operates normally.
XHWID	Crosshair cursor line width (bit 07). When this bit is 0, the crosshair cursor lines are one pixel wide. When this bit is 1, the lines are two pixels wide. The extra pixels are added to the right of and below the pixels which lie on the lines corresponding to the cursor X and Y positions.
XHCL1	Select crosshair clipping region (bit 06). If this bit is 1, region detector 1 is used to clip the crosshair cursor; if it is 0, region detector 2 is used. This bit is effective only if the crosshair cursor is selected (bit XHAIR is 1) and crosshair clipping is selected (bit XHCLP is 1).
XHCLP	Clip crosshair inside region (bit 05). If this bit is 1, the crosshair cursor is clipped so that it is displayed only within the region selected by the XHCL1 bit. If this bit is 0, the crosshairs extend to the edges of the displayed raster. This bit is effective only if the crosshair cursor is selected (bit XHAIR is 1).
XHAIR	Crosshair/sprite cursor select (bit 04). If this bit is 1, the cursor chip generates a crosshair whose lines intersect at the cursor X, Y position. If this bit is 0, the cursor chip generates the sprite pattern with its upper left corner at the cursor X, Y position.
FOPB	Force cursor plane B output to 1 (bit 03). When this bit is 1, the output from cursor plane B is forced to 1 throughout the display, regardless of the settings of bits ENPB, XHAIR, XHCLP, XHCL1, XHWID, and of the contents of the sprite plane B array. When this bit is 0, the cursor is displayed normally.

Data Bit	Definition
ENPB	Enable cursor plane B (02). When this bit is 0, the output from cursor plane B is inhibited; it is 0 throughout the display. When this bit is 1, the output from cursor plane B is displayed normally.
FOPA	Force cursor plane A output to 1 (bit 01). When this bit is 1, the output from cursor plane A is forced to 1 throughout the display, regardless of the settings of bits ENPA, XHAIR, XHCLP, XHCL1, XHWID, and of the contents of the sprite plane A array. When this bit is 0, the cursor is displayed normally.
ENPA	Enable cursor plane A (bit 00). When this bit is 0, the output from cursor plane A is inhibited; it is 0 throughout the display. When this bit is 1, the output from cursor plane A is displayed normally.

3.6.6 Loading the Cursor Sprite Pattern

The cursor sprite pattern is stored in two arrays, each made-up of sixteen 16-bit words. Each word of an array is displayed as 16 pixels on a scan line with bit 0 (least significant) in the leftmost display position. All 32 words are loaded by writing to the CUR_LOAD register. An internal address counter in the chip is incremented after each write to point to the next word in the array to be loaded.

Cursor command register bit LODSA controls access to the sprite arrays. When this bit is 0, the arrays are read during normal raster scanning to display the sprite pattern. When LODSA is 1, normal display of the sprite is inhibited and data can be written into the arrays. Changing LODSA from 1 to 0 resets the internal array address counter. The next write to CUR_LOAD loads the top line of the A plane array; the next fifteen writes load its remaining lines. The 16th through 32nd writes load the B plane array from top to bottom. When loading is completed, cursor command register bit LODSA must be reset to 0 to resume normal sprite display.

Loading the sprite arrays should be synchronized by waiting for the end-of-frame interrupt so that loading is done during the vertical blanking interval.

NOTE: *Only writes to CUR_LOAD advance the address counter. Any of the other registers of the cursor chip may be written to while the sprite arrays are being loaded.*

3.6.7 Cursor Region Detector

There are two region detectors, 1 and 2, each of which defines a rectangular area of the raster which can be used to clip the display of a crosshair cursor. Each region detector is programmed by setting four registers: CUR_XMIN, CUR_XMAX, CUR_YMIN, and CUR_YMAX. The horizontal boundaries of a region are controlled by the CUR_X... registers and can be specified only to a four-pixel boundary: the least significant two bits of their contents are ignored and the system behaves as if those two bits were always 0. The vertical boundaries are controlled by the CUR_Y... registers and can be specified to any line boundary. The offsets described in Section 3.6.2 must be applied to the values loaded into these registers.

The contents of the ...MIN registers determine the leftmost pixel or topmost line in a region. The contents of the ...MAX registers determine the first subsequent pixel or line which is no longer in the region. In other words, a ...MAX register should be loaded with the sum of the ...MIN value and the width or height of the region. The contents of a ...MAX register must always be greater than those of its corresponding ...MIN register.

3.6.8 Displaying a Sprite Cursor

A 16-by-16 pixel sprite cursor is displayed when cursor command register bit XHAIR is cleared to 0. The displayed position of the upper left corner of the sprite is controlled by the contents of the CUR_XPOS and CUR_YPOS registers. The values loaded into these registers must include an offset as described in Section 3.6.2. The cursor may be positioned at any pixel in both axes and may be positioned so that part of it falls outside the visible raster.

3.6.9 Displaying a Crosshair Cursor

A crosshair cursor is displayed when cursor command register bit XHAIR is set to 1. This cursor consists of a vertical line and a horizontal line which cross at the point determined by the contents of the CUR_XPOS and CUR_YPOS registers. The values loaded into these registers must include an offset as described in Section 3.6.2. The cursor may be positioned at any pixel in both axes.

Cursor command register bit XHWID controls the width of the lines. If XHWID is 0, the lines are 1 pixel wide. If XHWID is 1, the lines are doubled in width by adding another line one pixel to the right of the vertical line and below the horizontal line.

The length of the lines is controlled by cursor command register bit XHCLP. If XHCLP is 0, the lines extend the full width and height of the raster. If XHCLP is 1, the lines are clipped by the region detector selected by cursor command register bit XHCL1: a 1 in XHCL1 selects region 1 and a 0 selects region 2.

3.6.10 Controlling Cursor Plane Outputs

For each cursor plane (A and B), there are two bits in the cursor command register which control each plane's output, the enable bit and the force bit. The enable bit for plane A is ENPA and the enable bit for plane B is ENPB. If either of these is 1, normal cursor data (sprite or crosshair) is generated for the corresponding plane. If either of these is 0, the corresponding plane output is always 0. Setting both of these bits to 0 suppresses the cursor display so that the screen shows only the contents of the data plane. These bits are buffered so that they take effect only at the start of a vertical blanking interval.

The force bit for plane A is FOPA and the force bit for plane B is FOPB. If either of these is 1, the output of the corresponding plane is always 1 throughout the entire display raster regardless of the state of the plane's enable bit. The force bits are not buffered. They take effect immediately upon loading. These bits must be 0 for normal display operation.

3.6.11 Blanking the Display

The screen may be blanked without disturbing the display data plane or the cursor by using the cursor plane control bits to force the output of the B plane to 1 (set cursor command register bit FOPB) and the A plane to 0 (clear cursor command register bits FOPA and ENPA).

3.6.12 Cursor Chip Test

The cursor chip has a test flipflop which can be used to verify that the chip is functioning correctly. The state of this flipflop appears in bit 4 of the configuration and test register CFGTST. The value of this bit is the complement of the flipflop output, so a flipflop value of 0 appears as a 1 in bit 4 and vice versa.

To activate the test feature, cursor command register bit TEST must be cleared to 0. The test flipflop is cleared to 0 whenever the cursor command register is written to. The test flipflop is set to 1 by the logical OR of the outputs from cursor plane A, cursor plane B, region detector 1, and region detector 2.

Note that a test requires one full frame time to execute. A test procedure should wait for an end-of-frame interrupt, set up the test conditions, wait for another end-of-frame interrupt, write to the cursor command register to clear the test flipflop, wait for the next end-of-frame interrupt, and then look at the test flipflop value.

3.6.13 Power-Up Initialization

Power-up initialization sets the following to true.

- Controller select register VDC_SEL is 00h.
- End-of-frame interrupt is masked off.
- Display origin register VDC_ORG is 00h.
- Cursor chip register contents are indeterminate.
- Data plane storage contents are indeterminate.

The cursor chip requires two vertical blanking cycles to perform internal initialization before its registers can be loaded. To provide a clean appearance on the monitor, the startup code should wait for at least 50 milliseconds (for cursor chip internal initialization) and then set cursor command register bits TEST, HSHI, VBHI and FOPB to 1 and clear the others. This sets the proper sync signal polarity and blanks the screen by forcing the B plane output to 1 and the A plane output to 0.

NOTE: *The cursor command register bits TEST, HSHI, and VBHI must always be set to 1 for normal operation.*

3.7 Serial Line Controller (DZ Controller)

The system module serial line controller (Figure 3-56) handles four asynchronous serial lines. The heart of the controller is a DC367B gate array. Input characters from all four lines are buffered in a common 64-position silo. Only 1 line, the communication line, has full modem control signals. Figure 3-57 shows the DC376B gate array DZ controller and Table 3-22 lists the functions of the pins.

Figure 3-56: Serial Line Controller

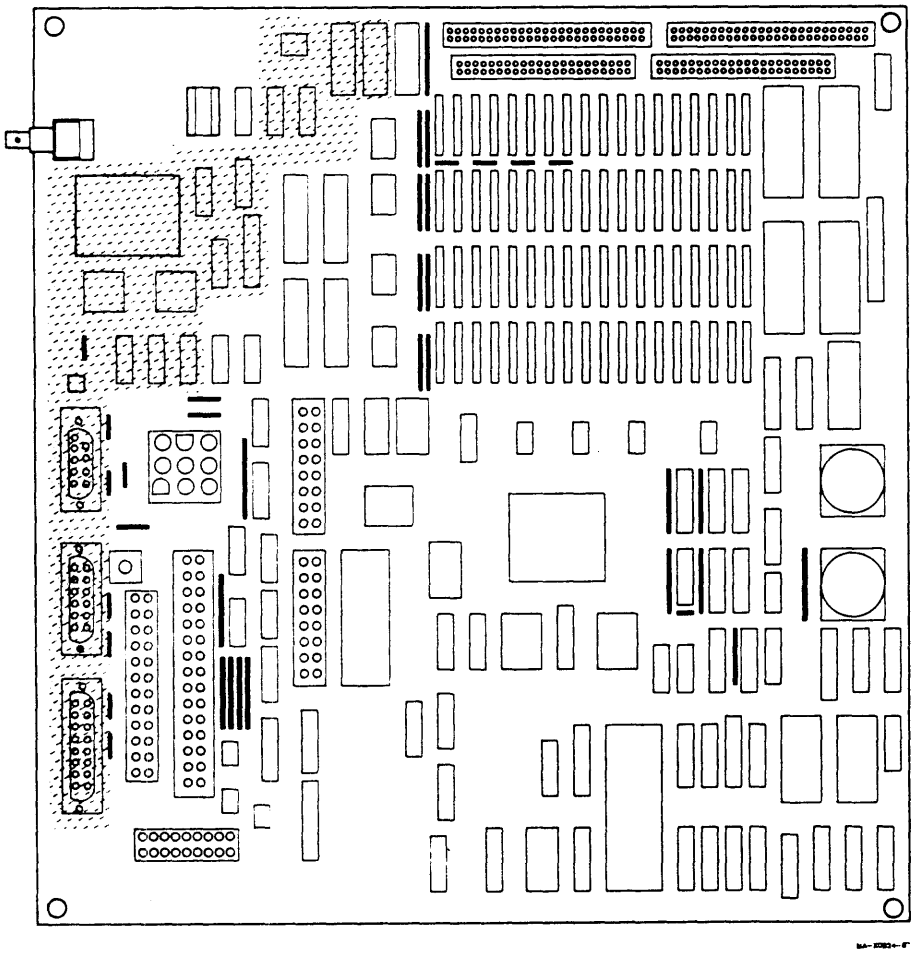


Figure 3–57: DZ Controller Chip Pinout

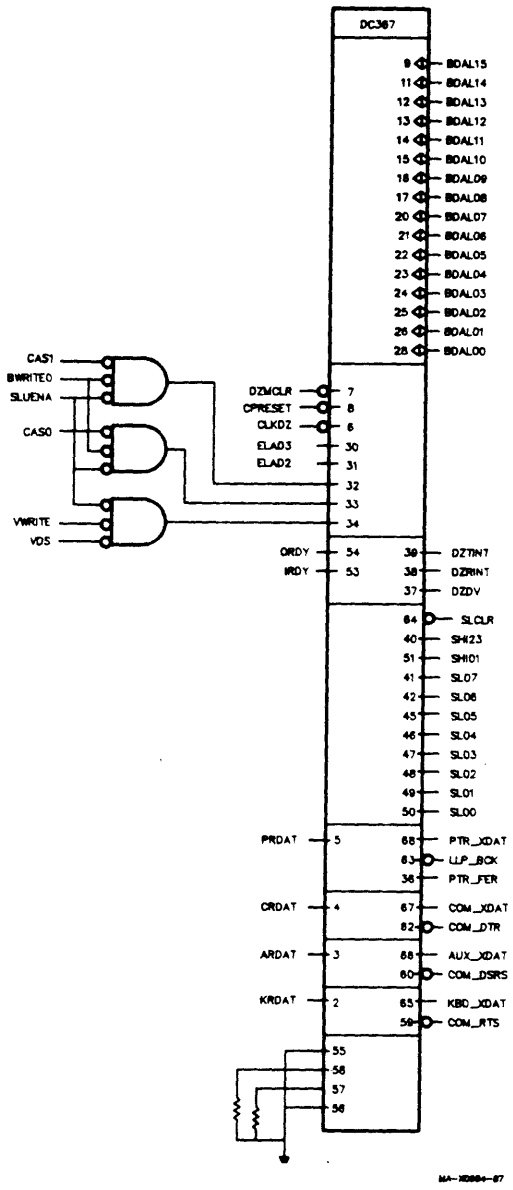


Table 3-22: DZ Controller Chip Pin Functions

Pin	Signal	Description
9 17:11 26:20 28	BDAL15 BDAL14:8 BDAL7:1 BDAL00	These signals are the address and data bus lines.
7	DZMCLR	This signal is the modem clear line.
8	CPRESET	This signal is the reset signal from the standard cell.
6	CLKDZ	This signal is the clock input from the 5.0688 MHz oscillator.
30:31	ELAD3:2	These signals are the latched address lines from the CPU chip.
32	WRHB	This signal is the logical AND of CAS1, BWRITE0, and SLUENA. They indicate when the valid high address byte is on the BDAL15:00 bus.
33	WRLB	This signal is the logical AND of CAS0, BWRITE0, and SLUENA. They indicate when the valid low address byte is on the BDAL15:00 bus.
34	RDEV	This signal is the logical AND of VWRITE, VDS, and SLUENA. They indicate when valid data is on the BDAL15:00 bus.
39	DZTINT	This signal is the transmit done interrupt line to the standard cell.
38	DZRINT	This signal is the receiver done or silo full interrupt line to the standard cell.
37	DZDV	This signal is the shift out signal. The DZ controller outputs this signal when the silo is ready to output a character. However, the silo does not output the character until the standard cell asserts the SHSILO signal.
54	ORDY	This signal is from the silo and indicates when it is ready to shift out a character.
53	INDY	This signal is from the silo and indicates when it is ready to receive another character.
64	SLCLR	This signal is the silo clear signal. It clears the silo when asserted.

Table 3–22 (Cont.): DZ Controller Chip Pin Functions

Pin	Signal	Description
40	SHI23	This signal is the shift in line to the silo. It indicates when the high byte silo must shift in a character.
51	SHI01	This signal is the shift in line to the silo. It indicates when the low byte silo must shift in a character.
41:42 45:50	SL07:06 SL05:00	These signals are the serial line character bus. They carry the input character to the silo.
5	PRDAT	This signal is the input data from the printer serial line (line 3).
4	CRDAT	This signal is the input data from the communication serial line (line 2).
3	ARDAT	This signal is the input data from the auxiliary, or pointer, serial line (line 1).
2	KRDAT	This signal is the input data from the keyboard, or console, serial line (line 0).
68	PTR_XDAT	This signal is the output data to the printer serial line (line 3).
67	COM_XDAT	This signal is the output data to the communication serial line (line 2).
66	AUX_XDAT	This signal is the output data to the auxiliary, or pointer, serial line (line 1).
65	KBD_XDAT	This signal is the output data to the keyboard, or console, serial line (line 0).

Table 3–22 (Cont.): DZ Controller Chip Pin Functions

Pin	Signal	Description
63	LLP_BCK	This signal is the local loopback modem control line. This line appears in the communication connector only.
62	COM_DTR	This signal is the data terminal ready modem control line. This line appears in the communication connector only.
60	COM_DSRS	This signal is the data signaling rate selector modem control line. This line appears in the communication connector only.
59	COM_RTS	This signal is the request to send modem control line. This line appears in the communication connector only.
36	PTR_FER	This signal indicates that the break key (halt) character has been received from the printer serial line. This halts the CPU when the BCC08 cable is connected to the printer port (the BCC08 cable shorts pins 8 and 9 which enable halts on this line.)

3.7.1 DZ Silo

The data is shifted into the silo in two bytes. The DZ chip controls which byte is enabled by the shift-in (SHI01 and SHI23) signals. SHI01 is shifted in first, then SHI23 is shifted in. It takes approximately a microsecond for the data to fall through the silo. The silo is a true silo where a character drops through all 64 words in the silo before it is latched at the output. The in-ready (IRDY) signal indicates that the input is ready for another byte and out-ready (ORDY) indicates that a byte has fallen through the silo and can be read at the output. Figure 3-58 shows the DZ silo.

3.7.2 Line Identification

The four serial lines on the serial line controller are numbered 0, 1, 2, and 3. Table 3-23 lists the use of each serial line.

Figure 3–58: DZ Silo Circuit Diagram

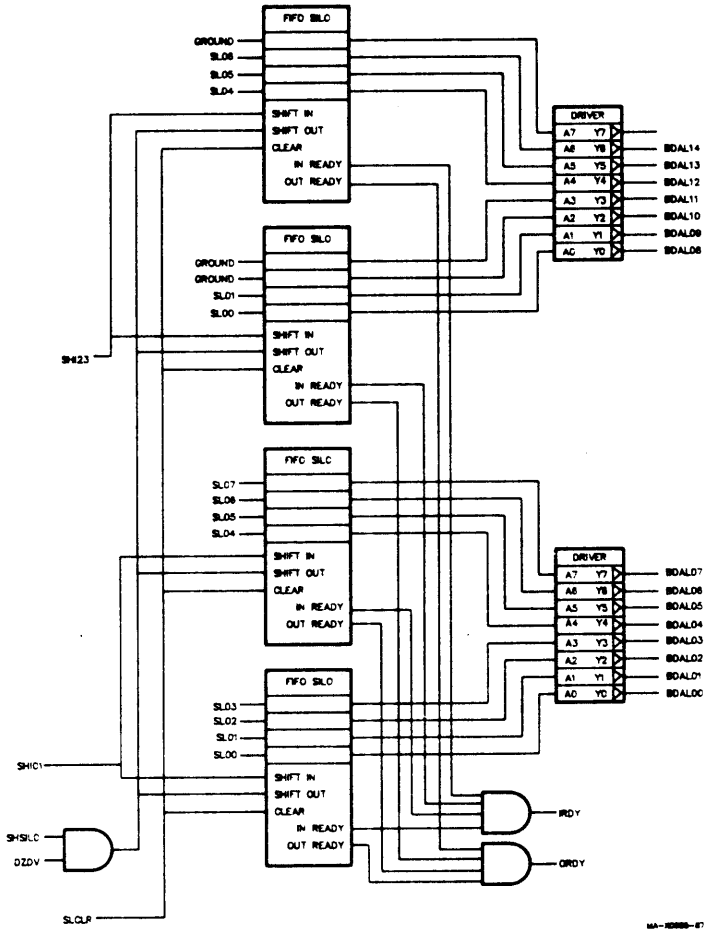


Table 3–23: Serial Line Identification

Line	Device	Definition
0	Keyboard	Connected to an LK201 keyboard through the video monitor cable. Data leads only. On the MicroVAX 2000 system, this line corresponds to port 1 on the DEC423 converter which is used for the console terminal.
1	Pointer	Connected to a VSXXX-AA mouse or VSXXX-AB tablet through the video monitor cable. Data leads only. On the MicroVAX 2000 system, this line corresponds to port 2 on the DEC423 converter which is used for a second terminal connection.
2	Communication	Connected to a 25-pin male D-shell connector for use with an external modem on both systems. Supports modem control signals DTR, RTS, RI, CD, DSR, CTS, DSRS, SPDMI, LLBK and TMI.
3	Printer	Connected to a 9-pin male D-shell connector for a serial printer. Data leads only. This line is also used to attach a diagnostic terminal to the system when using a special BCC08 cable. On the MicroVAX 2000 system, this line corresponds to port 3 on the DEC423 converter which is used for connection of a printer or a third terminal.

3.7.3 Diagnostic Terminal Connection

Line 3 on the VAXstation 2000 system is normally connected to a printer with a BCC05 cable. This line may instead be connected to a terminal for field service diagnostics by using a BCC08 cable. The BCC08 cable has a jumper between pins 8 and 9 on the 9-pin connector end of the cable. Bit L3CON of the configuration and test register CFGTST (see Section 3.11.2) is set to 1 when this jumper is present. Bit 3CON is 0 when the normal (BCC05) printer cable is used. When this jumper is present, a BREAK received on line 3 asserts the CPU HALT signal which causes a processor restart with restart code 02h (see Section 3.2.6.2). The MicroVAX 2000 system cannot use this diagnostic terminal since the DEC423 inhibits the connection of the BCC08 cable.

3.7.4 Interrupts

The controller generates two types of interrupt requests, each with a separate vector and bit in the INT_REQ and INT_MSK registers. These are transmitter done, and either receiver done or silo alarm. Section 3.5.9.4 lists the vector values. In order for these interrupts to be signalled to the CPU, the appropriate bits in the interrupt mask register INT_MSK must be set (See Section 3.5.9.2).

3.7.5 Register Summary

The serial line controller contains six addressable registers. Table 3-24 lists the six addressable registers.

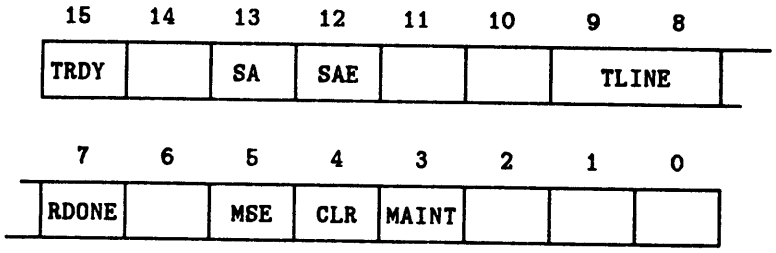
Table 3-24: Serial Line Controller Register Addresses

Address	Name	Access	Description
200A.0000	SERCSR	Read/write	Control and status register
200A.0004	SERRBUF	Read	Receiver buffer (bottom of silo)
200A.0004	SERLPR	Write	Line parameter register
200A.0008	SERTCR	Read/write	Transmitter control register
200A.000C	SERMSR	Read	Modem status register
200A.000C	SERTDR	Write	Transmitter data register

3.7.5.1 Control and Status Register (SER_CSR)

The control and status register is a 16-bit register at address 200A.0000. This register must be read on a word basis but can be written to on either a word or byte basis. All bits in SER_CSR are cleared to 0 by power-on or by setting the master clear bit CLR. Figure 3-59 shows the serial line control and status register.

Figure 3-59: Serial Line Control and Status Register (SER_CSR)



Data Bit	Definition
TRDY	Transmitter ready (bit 15). This read-only bit is set by the hardware when the transmitter scanner stops on a line whose transmitter buffer is ready to be loaded with another character and whose related transmitter control register SER_TCR's bit TXEN_x is set. The TLINE bits are only valid when the TRDY bit is 1. When TRDY changes from 0 to 1, the interrupt request register (INT_REQ Section 3.5.9.1) bit ST is also set to 1. If the interrupt mask register (INT_MSK) bit ST is also 1, then a transmitter interrupt request is sent to the CPU. Otherwise TRDY can be polled by the host program. However, the interrupt request register's bit ST is not automatically cleared while interrupts are masked, so when changing from polled to interrupt operation, there may be an interrupt request sent to the CPU unless the ST bit in INT_REQ is cleared by writing a 1 to the ST bit in INT_CLR. The TRDY bit is cleared when data is loaded into the transmitter for the line number indicated in TLINE by writing to register SER_TDR. If additional transmitter lines need service, TRDY is set again within 1.4 microseconds of the completion of the transmitter data load operation. The TRDY bit is also cleared when the master scan enable bit MSE is cleared, or when the related transmitter control register (SER_TCR) bit TXEN_x is cleared.
14	Not used.

Data Bit	Definition
SA	Silo alarm (bit 13). This read-only bit is set by the hardware when 16 characters have been entered into the FIFO silo buffer. While the silo alarm enable bit SAE is 1, the transition of SA from 0 to 1 sets interrupt request register (INT_REQ) bit SR to 1. If interrupt mask register bit SR is also 1, an interrupt is sent to the CPU. Otherwise the SA bit may be polled. However, the interrupt request register bit SR is not automatically cleared while that interrupt is masked, so when changing from polled to interrupt operation, there may be an interrupt request to the CPU unless the host program clears SR by writing a 1 to the interrupt clear register (INT_CLR) bit SR. The SA bit is cleared by reading the receiver buffer register SER_RBUF. When responding to a silo alarm, the host program reads characters from the silo until it is empty (until DVAL in register SER_RBUF is 0), since the silo alarm bit is not set again until 16 additional characters have been stored in the silo. The SA bit is always 0 while the silo alarm enable bit SAE is 0.
SAE	Silo alarm enable (bit 12). This read/write bit selects the source of the receive interrupt request signal. If SAE is 1, the silo alarm bit SA is used as the signal. If SAE is 0, the receiver done bit RDONE discussed below is used instead.
11:10	Not used.
TLINE	Transmitter line number (bits 9:8). These read-only bits indicate the number of the line whose transmitter buffer needs servicing (bit 8 is the least significant bit). These bits are only valid while the transmitter ready bit TRDY is 1.

These bits are cleared when the master-scan enable bit MSE is cleared.

RDONE	Receiver done (bit 7). This read-only bit is set by the hardware when an incoming character appears at the output of the silo buffer. While the silo alarm enable bit SAE is 0, the transition of RDONE from 0 to 1 sets interrupt request register (INT_REQ) bit SR to 1. If interrupt mask register (INT_MSK) bit SR is also 1, an interrupt is signalled to the CPU. Otherwise the RDONE bit may be polled. However, the interrupt request register (INT_REQ) bit SR is not automatically cleared while that interrupt is masked, so when changing from polled to interrupt operation, there may be an interrupt request sent to the CPU unless the interrupt request register (INT_REQ) bit SR is cleared by writing a 1 to the interrupt clear register (INT_CLR) bit SR.

Data Bit	Definition
	RDONE is cleared when the receiver buffer register SER_RBUF is read. If another character is available in the silo, RDONE is set again after a delay of between 0.1 and 1.0 microseconds. This bit is also cleared when the master scan enable bit MSE is cleared.
6	Not used.
MSE	Master scan enable (bit 5). This read/write bit must be set to 1 to permit the receiver and transmitter control sections to scan the lines to see if they need servicing. When this bit is 0, the transmitter ready bit TRDY is cleared and the receiver silo is cleared.
CLR	Master clear (bit 4). When this bit is set, the system performs an internal initialization process. At the conclusion of this process the system clears this bit. If this bit is 1, then the internal process is not complete. This initialization clears all registers, the silo, and all UARTs, but there are some exceptions as noted below. In the receiver buffer register (SER_RBUF), only bit DVAL is cleared. The remaining bits are not affected. Bits 15:8 of the transmitter control register (SER_TCR modem control outputs) are not cleared. The modem status register (SER_MSR) is not cleared.

NOTE: *After setting the master clear bit CLR, a program must repeatedly read SER_CSR until it finds CLR equal to 0 before attempting any other operations with the serial line controller.*

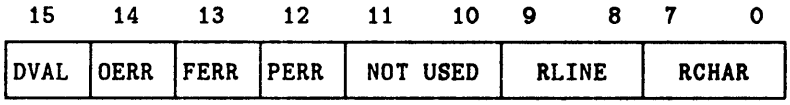
Neither of the interrupt controller registers (INT_REQ or INT_MSK) are altered when CLR is set. Bits SR and ST of INT_MSK and also INT_REQ must be cleared to 0 by writing 1s to the same bits of INT_CLR to complete the initialization process.

MAINT	Maintenance (bit 3). This read/write bit, when set, loops the serial output connections of the transmitters to the corresponding serial input connections of the receivers. This feature is intended for hardware diagnostic use.
2:0	Not used.

3.7.5.2 Serial Line Receiver Buffer Register (SER_RBUF)

The receiver buffer register is a 16-bit read-only register at address 200A.0004. It must be read as a word. It contains the received character at the bottom of the silo buffer (the oldest character in the silo). Reading this register removes the character from the silo buffer, and all the other characters in the silo are shifted down to the lowest unoccupied location. When this register is read (or when the master clear bit CLR in SER_CSR is set or after a power-on reset), the data valid bit DVAL in SER_RBUF is cleared and the remaining bits of the register (although not cleared) are invalid. Figure 3-60 shows the serial line receiver buffer register.

Figure 3-60: Serial Line Receiver Buffer Register (SER_RBUF)



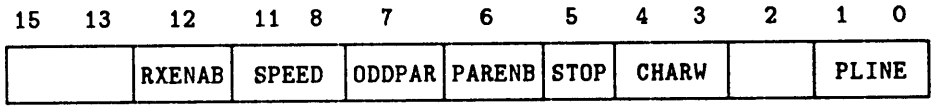
Data Bit	Definition
DVAL	Data valid (bit 15). This bit, when 1, indicates that the data in bits 14:0 of the register are valid. This permits an interrupt handling program to read the receiver buffer register repeatedly and store each character until this bit is read as 0, which indicates that the silo is empty.
OERR	Overrun error (bit 14). This bit is 1 when a received character is overwritten in a UART buffer by a following character before the first character was transferred to the silo. This condition indicates that the program is not emptying the silo fast enough.
FERR	Framing error (bit 13). This bit is 1 if the received character did not have a stop bit present at the correct time. The combination of FERR set and RCHAR entirely 0 is usually interpreted as indicating that a BREAK has been received. The receipt of a framing error on line 3 (the printer port) is a special case. If the hardware detects a framing error on line 3 and the accompanying character contains all 0s (i.e. a BREAK has been received), the line controller hardware asserts a signal whose effect is described under Section 3.7.3 Diagnostic Terminal Connection.
PERR	Parity error (bit 12). This bit is 1 if the sense of the parity of the accompanying character does not agree with the parity which was defined for the line when its line parameter register SER_LPR was last loaded.
11:10	Not used.

Data Bit	Definition
RLINE	Receiver line number (bits 9:8). These bits indicate the number of the line from which the character was received (bit 8 is the least significant bit).
RCHAR	Received character (bits 7:0). Characters with a width of fewer than 8 bits (as defined when the line's line parameter register was last loaded) are right justified with the unused bit positions cleared. The parity bit is not included in the received character.

3.7.5.3 Serial Line Parameter Register (SER_LPR)

The line parameter register is a 16-bit register at address 200A.0004 that controls the operating parameters of each line. This register is write-only and must be written as a 16-bit word. The parameters for each line must be reloaded after each power-on reset or setting of the master clear bit CLR in SER_CSR. The operating parameters should not be modified for a line while data transmission or reception is in progress on that line. Figure 3-61 shows the serial line parameter register.

Figure 3-61: Serial Line Parameter Register (SER_LPR)



Data Bits	Definition
15:13	Not used.
RXENAB	Receiver enable (bit 12). This bit must be set in order for the UART for this line to receive bits and assemble them into characters.
SPEED	Speed code (bits 11:8). These bits select the data bit rate for the receiver and transmitter for the line. The bits are encoded as follows.

11	10	9	8	Data Rate (Bits/Second)
0	0	0	0	50
0	0	0	1	75
0	0	1	0	110
0	0	1	1	134.5
0	1	0	0	150
0	1	0	1	300
0	1	1	0	600
0	1	1	1	1200
1	0	0	0	1800
1	0	0	1	2000
1	0	1	0	2400
1	0	1	1	3600
1	1	0	0	4800
1	1	0	1	7200
1	1	1	0	9600
1	1	1	1	19200

Data Bits	Definition
ODDPAR	Odd parity (bit 7). If this bit is set and the parity enable bit PARENB in SER_LPR is also set, then characters with odd parity are transmitted to the line and characters received from the line are expected to have odd parity. If this bit is clear and the parity enable bit PARENB in SER_LPR is set, then characters with even parity are transmitted to the line and characters received from the line are expected to have even parity. If the parity enable bit PARENB in SER_LPR is clear, then the setting of this bit is immaterial.
PARENB	Parity enable (bit 6). If this bit is set, characters transmitted to the line have a parity bit appended and characters received from the line have their parity checked. The sense of the parity is according to the setting of the odd parity bit ODDPAR in SER_LPR.
STOP	Stop code (bit 5). If this bit is clear, the stop code following the last transmitted bit is 1 bit time long. If this bit is set, the stop code lasts 1.5 bit times for characters whose width is 5 bits, and 2 bit times for characters whose width is 6, 7 or 8 bits.
CHARW	Character width (bits 4:3). These bits control the number of data bits (exclusive of any parity bit) in the characters transmitted and expected in the characters received. The encoding is below.

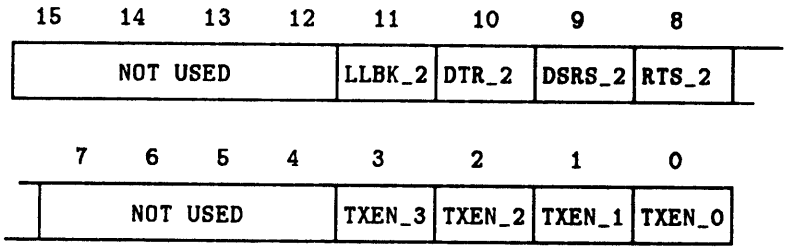
4	3	Character Width (Bits)
0	0	5
0	1	6
1	0	7
1	1	8

2	Not used.
PLINE	Parameter line number (bits 1:0). These bits specify the number of the line to which the parameters in the rest of the register apply. Bit 0 is the least significant bit.

3.7.5.4 Serial Line Transmitter Control Register (SER_TCR)

The transmitter control register is a 16-bit register at address 200A.0008 that must be read on a word basis and can be written on either a word or byte basis. Figure 3-62 shows the serial line transmitter control register.

Figure 3-62: Serial Line Transmitter Control Register (SER_TCR)



Data Bits	Definition
15:12	Not used.
LLBK_2	Local loopback (bit 11). This read/write bit controls the state of the local loopback modem control signal (CCITT circuit 141) for line 2. Setting the bit asserts the ON state of the LLBK signal. This bit is cleared by a power-on reset; it is NOT cleared when the master clear bit CLR in SER_CSR is set.
DTR_2	Data terminal ready (bit 10). This read/write bit controls the state of the data terminal ready modem control signal (CCITT circuit 108/2) for line 2. Setting the bit asserts the ON state of the DTR signal. This bit is cleared by a power-on reset; it is NOT cleared when the master clear bit CLR in SER_CSR is set.
DSRS_2	Data signalling rate selector (bit 9). This read/write bit controls the state of the data signalling rate selector modem control signal (CCITT circuit 111) for line 2. Setting the bit asserts the ON state of the DSRS signal. This bit is cleared by a power-on reset; it is NOT cleared when the master clear bit CLR in SER_CSR is set.
RTS_2	Request to send (bit 8). This read/write bit controls the state of the request to send modem control signal (CCITT circuit 105) for line 2. Setting the bit asserts the ON state of the RTS signal. This bit is cleared by a power-on reset; it is NOT cleared when the master clear bit CLR in SER_CSR is set.
7:4	Not used.

Data Bits	Definition
TXEN_x	Transmitter line enable (bits 3:0). These read/write bits enable the transmitter logic for lines 3, 2, 1, and 0, respectively. Setting each of these bits causes the transmitter scanner to stop and assert the transmitter ready bit TRDY in SER_CSR if the UART for that line has a transmitter buffer empty condition. The transmitter scanner resumes scanning when either the transmitter data register for the line at which the scanner stopped is loaded with another character, or when that line's transmitter line enable bit is cleared. A transmitter line enable bit should only be cleared while the scanner is not running (i.e. when the transmitter ready bit TRDY in SER_CSR is set or the master scan enable bit MSE in SER_CSR is clear). The transmitter line enable bits are cleared by a power-on reset and whenever the master clear bit CLR in SER_CSR is set.

3.7.5.5 Modem Status Register (SER_MSR)

The modem status register is a 16-bit read-only register at address 200A.000C which contains the status of modem input signals for line 2. The ON condition of a modem signal is presented as the set state of the corresponding bit. Figure 3-63 shows the serial line modem status register.

Figure 3-63: Serial Line Modem Status Register (SER_MSR)

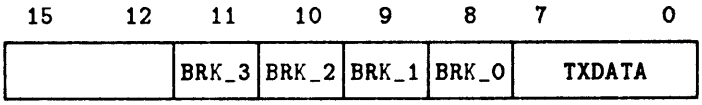
15 12	11	10	9	8	7 4	3	2	1	0
	SPDI_2	CD_2	DSR_2	CTS_2		0	RI_2	0	TMI_2

Data Bits	Definition
15:12	Not used; read values undefined.
SPDI_2	Speed mode indicate (bit 11). This bit reflects the state of the speed mode indicate signal from an external modem (CCITT circuit 112) on line 2. The set state corresponds to the ON state of the signal.
CD_2	Carrier detect (bit 10). This bit reflects the state of the carrier detect signal from an external modem (CCITT circuit 109) on line 2. The set state corresponds to the ON state of the signal.
DSR_2	Data set ready (bit 9). This bit reflects the state of the data set ready signal from an external modem (CCITT circuit 107) on line 2. The set state corresponds to the ON state of the signal.
CTS_2	Clear to send (bit 8). This bit reflects the state of the clear to send signal from an external modem (CCITT circuit 106) on line 2. The set state corresponds to the ON state of the signal.
7:4	Not used; read values undefined.
3	Reserved, reads as 0.
RI_2	Ring indicator (bit 2). This bit reflects the state of the ring indicator signal from an external modem (CCITT circuit 125) on line 2. The set state corresponds to the ON state of the signal.
1	Reserved, reads as 0.
TMI_2	Test mode indicate (bit 0). This bit reflects the state of the test mode indicate signal from an external modem (CCITT circuit 142) on line 2. The set state corresponds to the ON state of the signal.

3.7.5.6 Transmitter Data Register (SER_TDR)

The transmitter data register is a 16-bit write-only register at address 200A.000C. It can be written on either a word or byte basis. Figure 3-64 shows the serial line transmitter data register.

Figure 3-64: Serial Line Transmitter Data Register (SER_TDR)



Data Bits	Definition
15:12	Not used.
BRK_x	Break control (bits 11:8). These write-only bits control the assertion of a BREAK condition on lines 3, 2, 1, and 0, respectively. Setting a bit immediately forces the transmitter output for the corresponding line to the SPACE condition. This condition will persist until the break control bit is cleared. These bits are cleared by a power-on reset and when the master clear bit CLR in SER_CSR is set.
TXDATA	Transmitter buffer (bits 7:0). Data to be transmitted by a line's UART is loaded into these 8 bits. If the character width is less than 8, the unused bits are at the high-order (bit 7) end of the byte. This register may be written to only while the transmitter ready bit TRDY in SER_CSR is set. The line to which the character is sent is indicated by the transmitter line number bits TLINE in SER_CSR.

3.8 9224 Disk Controller

This section describes the 9224 disk controller (Figure 3-65).

The disk controller supports both diskette drives (RX33) and ST506/412 hard disk drives (RD32 and RD53). The maximum configuration of the controller is one diskette drive and two hard disk drives. The controller is an HDC 9224 universal disk controller chip which uses a phase-locked loop data recovery circuit, an address counter, and a 16 Kbyte dual port data buffer. Figure 3-66 shows the pinout of the 9224 disk controller chip and Table 3-25 lists the signals for each pin on the 9224 disk controller chip.

Figure 3-65: 9224 Disk Controller

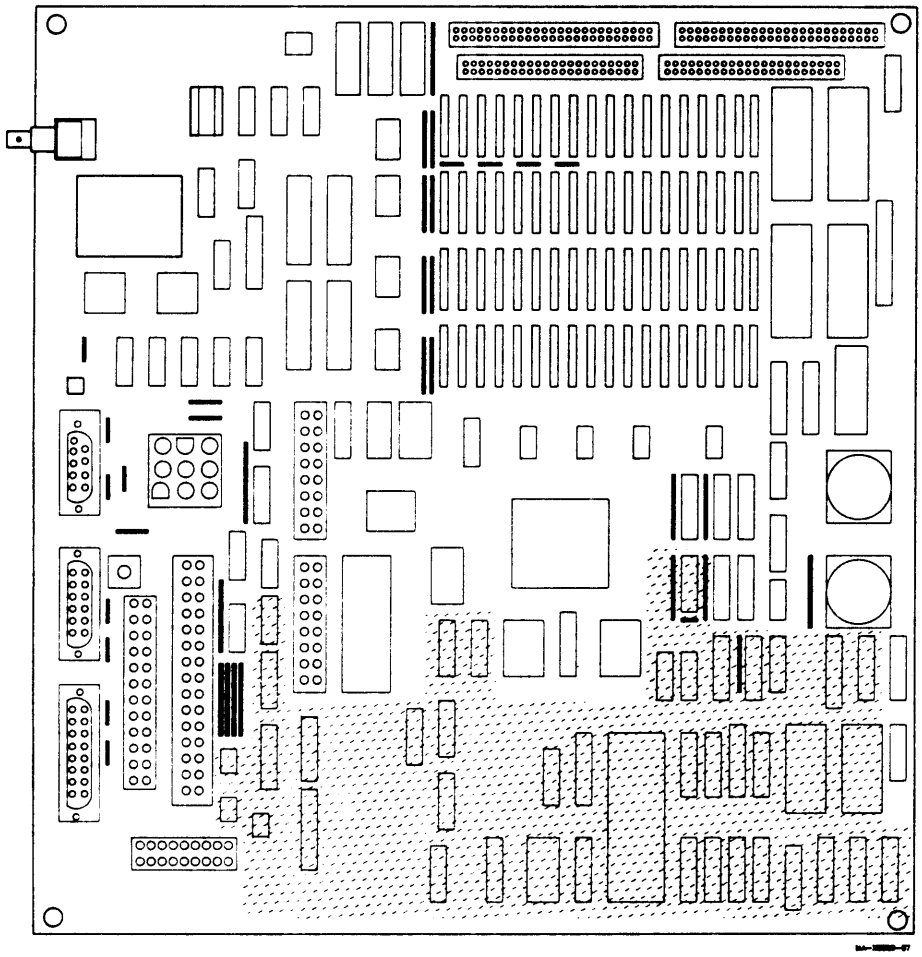


Figure 3-66: 9224 Disk Controller Chip Pinout

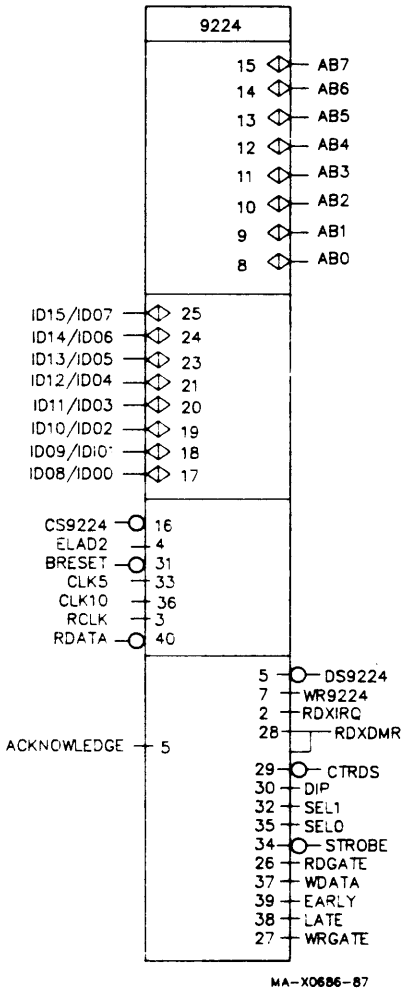


Table 3–25: 9224 Disk Controller Pin Description

Pin	Signal	Description
21:17 25:23	DB4:0 DB7:5	These signals are the data bus for the disk controller. They are connected directly to the low byte for the internal data bus (ID00:07) and indirectly through an eight bit transceiver to the high byte of the internal data bus (ID08:15). This bus transfers data to and from the disk data buffer and also to and from the CPU BDAL bus.
15:8	AB7:0	These signals are the auxiliary bus for the disk controller. They update the registers that contain information on the head select, drive select, step, and drive status information.
16	CS9224	This signal is the disk controller's chip select signal from the standard cell.
4	ELAD2	This signal is the low bit of the latched address bus from the VDAL bus (VDAL02). It is used during CPU and disk controller communication where a low indicates that data may be written to or read from the controllers internal registers and a high indicates that the CPU can write commands to or read command results from the controller.
31	BRESET	This signal is the reset signal from the standard cell. This signal resets the dick controller without having to power-down the entire system.
33	CLK5	This signal is the 5-MHz DMA clock from the standard cell.
36	CLK10	This signal is the 10-MHz disk controller clock from the standard cell.
3	RCLK	This signal is the read clock strobe from the standard cell. It acts as a window to indicate raw data cell boundaries on the RDDATA line.
40	RDDATA	This signal is the read data signal. It receives the raw read data from the disk through the standard cell's phase-locked loop recovery circuit.
6	DS9224	This signal is the data strobe for the disk controller. It is used by the disk controller or by the standard cell to indicate when valid data is available on the data bus (DB7:0) during a transfer to or from the disk data buffer.
7	WR9224	This signal is the read/write signal for the disk controller. It is used by the disk controller or by the standard cell to indicate if the disk data buffer transfer cycle is a read or write cycle cycle.

Table 3–25 (Cont.): 9224 Disk Controller Pin Description

Pin	Signal	Description
2	RDXIRQ	This signal is the interrupt request signal that is sent to the standard cell's interrupt controller when the disk controller needs service.
28	RDXDMR	This signal is the DMA request line. It is wrapped around and input directly to the DMA acknowledge line.
5	RDXDMR	This signal is the DMA acknowledge line. It is from the DMA request line on pin 28 of the disk controller.
29	ECCTIM	This signal is used with the DS9224 and DIP signals to increment the address counters.
30	DIP	This signal is the DMA is progress flag. It is active whenever the disk controller is performing a DMA operation.
32 35	SEL1 SEL0	These lines select one of the four control lines that enable registers on the AB7:0 bus. These select lines are decoded by a 2-4 decoder that is enabled by the STB signal.
6	STB	This signal is the strobe signal which enables a decoder that allows the selection of control lines to the registers on the AB7:0 bus.
26	RDGATE	This signal is the read gate strobe. It is used to start the reading cycle. It switches the voltage-controlled oscillator from locking onto the natural clock frequencies to locking onto the raw data off the disk.
37	WDATA	This signal is the data to be written to the disk.
38 39	LATE EARLY	These signals are the select lines for a write precompensation delay line multiplexer. The delay line is not used for the RX33, RD32, or the RD53 drives.
27	WRGATE	This signal is the write enable signal to the drives.
1	VCC	This is the +5 Vdc power connection to the disk controller.
22	VSS	This is the ground connection to the disk controller.

3.8.1 Disk Data Buffer

The disk data buffer is a 16K byte block of RAM storage that is shared between the disk controller, the tape controller, and the CPU. This buffer uses two 8K byte by 8-bit static RAM chips and is not included as part of the system module dynamic RAM. It is accessible to the CPU in all read and write access modes (byte, word and longword) and it occupies physical addresses 200D.0000 through 200D.3FFF.

The disk controller chip accesses this buffer using its built-in 24-bit DMA hardware when transferring data to and from a disk. To the tape controller, which generates a 24-bit DMA address, the data buffer is a byte-addressed block with an address range of 000000h through 003FFFh. The disk and tape controller access the data buffer through the address counters and the CPU accesses the data buffer through the ELAD9:2 bus and the MEMAD3:0 bus. The disk data buffer is accessed by the CPU chip through the tri-state transceivers between the BDAL bus and the IDAL bus. Only one controller can access the disk data buffer at one time. The device driver software must ensure that only one device at a time attempts to access the buffer. Figure 3-67 shows the circuit diagram of the data buffer.

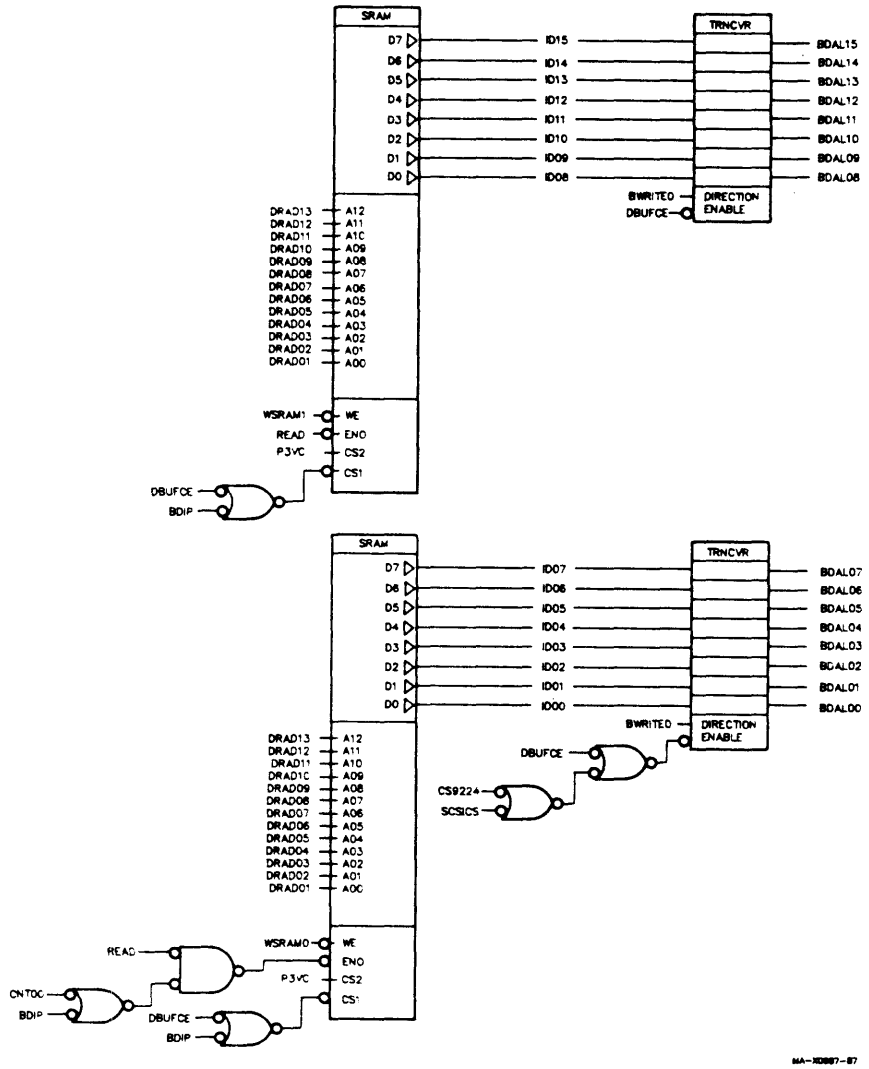
When the CPU is performing a write to the data buffer, the low 16 bits of data go directly to the buffer on the internal data bus (ID15:00) during the first half of the cycle and the high 16 bits are latched inside the standard cell. During the second half of the cycle, the latched high word is put on the ID15:00 bus to the buffer.

When the CPU is performing a read to the data buffer, the high 16 bits from the data buffer are addressed first and they are latched in the standard cell during the first half of the cycle. During the second half of the cycle, the low 16 bits from the data buffer are output onto the low byte of the data bus and the latch high bits are put on the high byte bus of the data bus at the same time to form the full 32-bit wide data bus.

3.8.2 Disk Address Counters

The address counters hold the data buffer address from the disk controller during normal RAM cycles as well as during DMA cycles. The disk controller uses a 24-bit DMA address and the system only uses a 16-bit address so the high byte is not used. The dropping of the high byte is done by emitting the high address byte onto the AB7:0 bus which is loaded into the first address counter. The middle address byte is then put on the AB7:0 bus next and is also loaded into the first address counter which pushes the high byte that was originally in the first counter into the second address counter. Finally, the low address byte is put on the AB7:0 bus and is loaded into the first address counter which pushes the middle byte into the second address counter. Since the system only uses a 16-bit address, a third address counter is not available and the high address byte is lost.

Figure 3-67: Disk Data Buffer Circuit Diagram

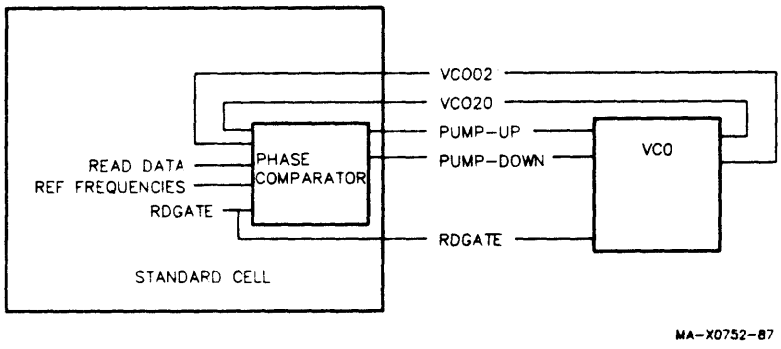


This leaves the low and middle address bytes in the address counters. The address counters then put the 16-bit address onto the CNT15:00 bus. The address is then sent through a multiplexer to the data buffer. The multiplexer channels the disk/tape address to the data buffer or it channels the CPU address to the data buffer when the CPU is accessing the data buffer. The disk controller sets up the multiplexer, control registers, and the high byte drop automatically at the start of a read/write operation.

3.8.3 Phase-Locked Loop

The phase-locked loop consists of a phase comparator and a voltage-controlled oscillator (VCO) as shown in Figure 3-68. The phase comparator is inside the standard cell. The VCO is a dual oscillator chip for both hard disk and floppy diskette data frequencies. The phase lock loop is used to control the frequency of the raw read data from the disks. The individual modified frequency modulation (MFM) pulses that are read from the disks are very sensitive to speed variations and the value of the pulse (1 or 0) may be lost if the frequency of the data stream is not precise. The VCO allows tracking of any variation of the data stream and sends feedback to the phase comparator to compensate the variation so the loop recovers the data and sends the disk controller a steady and reliable data stream.

Figure 3-68: Phase-Locked Loop Block Diagram



When the phase-locked loop is running but not reading from the disk, we lock it to a reference frequency generated within the standard cell. This reference frequency prevents the loop from drifting off to a very high or very low frequency when not reading data. If it did drift off then there would be a long delay time to get the loop back to the proper frequency before the system could read data from the disks. The reference frequency for the hard disks is 10 megahertz. The reference frequency for the floppy diskettes is 500 kilohertz when RX50 media is selected or 1 megahertz when RX33 media is selected.

When the phase-locked loop is reading data from the disks, the reference frequency for hard disks is 20 megahertz and for the floppy diskette is 2 megahertz. The VCO is a dual oscilator but only produces one frequency at a time. That is, either the hard disk reference frequency or the floppy diskette reference frequency.

3.8.3.1 Phase Comparator

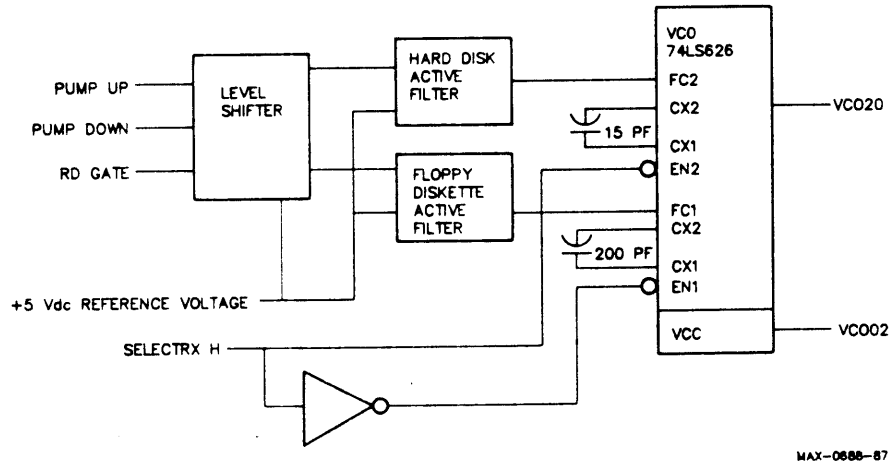
The phase comparator is internal to the standard cell. It has two 4-position multiplexers as its input and the output is the pump-up and pump-down signals to the VCO. Both input multiplexers have a 10 megahertz, 1 megahertz, and a 500 kilohertz reference frequency and one multiplexer has the output of the VCO as the fourth input and the other has the raw data from the disks as the fourth input. These multiplexers are controlled by the read gate (RDGATE) signal. When RDGATE is not asserted, the reference frequencies are allowed through the multiplexers to the comparator. When the disk controller starts a disk read operation, it asserts RDGATE which allows the output of the VCO and the raw read data from the disk to pass through to the comparator. The comparator consists of two edge catcher flip-flops whose clock input is the output of the multiplexers. The output of the flip-flops are the pump-up and pump-down signals that go to the VCO. As soon as the edge of the either input signal is received in the flip-flop, it is output to the VCO. A reset signal automatically resets the flip-flops to there original state before a second edge is received. These pump-up and pump-down signals should be identical. If they are identical, then the VCO does not change its output frequency. If the two pump signals are not identical, then the VCO increases or decreases frequency its output frequency to compensate for the difference.

3.8.3.2 Voltage-Controlled Oscillator (VCO)

The voltage-controlled oscillator is a 74LS626 dual oscillator. The VCO chip uses a level shifter circuit and an active filter circuit to provide accurate input signals. These VCO front end circuits input and integrate the pump-up and pump-down signals. They measure the amount of pulse width in each signal, sum them together, and send a phase-error signal voltage to the VCO chip to shift up or shift down the reference center frequency depending on the phase error signal. The output of the VCO chip is looped back to the phase comparator inside the standard cell.

The VCO chip has two output signals, one signal for the hard disk phase-locked loop data recovery circuit and the other for the floppy diskette phase-locked loop data recovery circuit. Only one of these output signals can be active at the same time. The output frequencies are determined by the value of a capacitor connected to the CX1 and CX2 input lines for each output signal. The enable for hard disk half of the VCO chip is the SELECTRX H signal and the floppy diskette half is the invert of the SELECTRX H signal (SELECTRX L) so that only one of the outputs is enabled at one time. The VCO circuit is powered by a special +5 Vdc that is divided down from the +12 Vdc supply. The VCO circuit uses this +5 Vdc for the reference voltages needed in the analog level shifter circuits and the analog active filter circuits. The level shifter is controlled by the RDGATE signal which indicates whether the phase-locked loop is locked on the reference frequencies or is locked on the raw read data. The +5 Vdc reference voltage is further divided to produce another reference (+3 Vdc voltage) supply for the VCO circuit. Figure 3-69 shows the block diagram of the VCO circuit.

Figure 3-69: VCO Block Diagram



3.8.4 Hard Disk Data Bus

The hard disk data bus contains the disk control signals such as the head select, drive select, and head positioning information as well as the raw data read and write signals. The disk controller uses the auxiliary bus (AB7:0) to transfer the control data to and from the disk. Raw data written to the disk is output on the disk controller's WDATA pin. Raw data read from the disk is processed through the phase-locked loop and then is presented to the disk controller on the RDATA pin. The system supports two hard disk drives. Both drives share the write data signal but each has a separate read data path. The shared write data signal is sent to the drive that is selected by the drive select signal.

3.8.5 Floppy Disk Data Bus

The floppy diskette data bus contains the floppy drive control signals such as the head select, head positioning information, and the high density select signal which indicates whether the media is RX50 or RX33 media. The disk controller uses the auxiliary bus (AB7:0) to transfer the control data to and from the floppy drive. Raw data written to the floppy drive is output on the disk controller's WDATA pin. Raw data read from the disk is processed through the phase-locked loop and then is presented to the disk controller on the RDATA pin. The system supports only one floppy diskette drive and that drive must be located in the system box.

3.8.6 Controller Chip Organization

The HDC 9224 controller chip has 15 internal registers which control its operation and reveal its status. These are indirectly accessible to the CPU by way of three ports that are mapped into the processor's address space. To help prevent confusion, the three ports that a program can access directly are named with the prefix DKC_, and the controller registers (which are accessible only via the ports) are named with the prefix UDC_.

The controller chip has an internal register pointer which designates the register which is accessible to the CPU via the register data access port. This pointer can be set explicitly by a SET REGISTER POINTER command written to the controller command port. It is implicitly incremented by accesses to the register data access port until it reaches the highest-numbered register (the UDC_DATA register), after which the pointer value continues to point to UDC_DATA until another SET REGISTER POINTER command is issued.

3.8.6.1 Disk Controller Chip Ports

Program access to the controller chip is via three 8-bit ports, each of which appears as the low-order byte of a longword address. Note that the command and status ports have the same address: one port is write-only and the other is read-only. Table 3-26 lists the address and access of the disk controller chip ports.

Table 3-26: Disk Controller Chip Ports

Address	Access	Name
200C.0000	Read/write	DKC_REG register data access
200C.0004	Write only	DKC_CMD controller command
200C.0004	Read only	DKC_STAT interrupt status

NOTE: *Consecutive accesses to controller chip ports must be separated by at least 0.7 microseconds, regardless of whether the accesses are reads or writes and of whether the same or different ports are designated.*

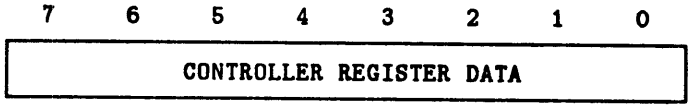
A program must not attempt to read or write any of the disk controller chip ports (nor any of the tape controller chip ports) while the controller is executing any data transfer command (e.g. any of the READ, WRITE, or FORMAT commands). This limitation is because the data path to the ports is also used by the controller to access the disk data buffer.

3.8.6.1.1 Disk Register Data Access Port

The register data access port is an 8-bit read/write port accessible to the CPU at physical address 200C.0000. This port provides CPU access to the controller register designated by the controller's internal register pointer. These registers are described below. Figure 3-70 shows the disk register data access port.

NOTE: *Some registers are read/write and others are read-only or write-only. In the latter two cases, a given value in the register pointer designates different registers depending upon whether the access to DKC_REG is a read or a write. In either case, each read or write access to DKC_REG advances the internal register pointer after the access is complete (until the pointer reaches the highest register number, 0Ah, after which it remains at that value). Therefore, CPU instructions which perform more than one access (such as BISB2 and BICB2) may not be used.*

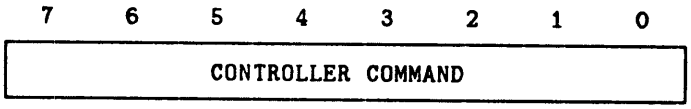
Figure 3-70: Disk Register Data Access Port



3.8.6.1.2 Disk Controller Command Port (DKC_CMD)

The controller command port is an 8-bit write-only port accessible to the CPU at physical address 200C.0004. The CPU instructs the disk controller to perform some action by writing a command byte to this port. Figure 3-71 shows the disk controller command port.

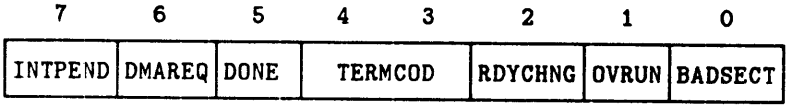
Figure 3-71: Disk Controller Command Port (DKC_CMD)



3.8.6.1.3 Interrupt Status Port (DKC_STAT)

The interrupt status port is an 8-bit read-only port accessible to the CPU at physical address 200C.0004. Figure 3-72 shows the disk interrupt status port.

Figure 3-72: Interrupt Status Port (DKC_STAT)



Data Bit	Definition
INTPEND	Interrupt pending (bit 7). This bit reflects the state of the hardware interrupt signal sent from the controller chip to the system's interrupt controller. The transition of this bit from 0 to 1 generates an interrupt request in the interrupt controller. The INTPEND bit is set to 1 in either of two cases: (1) when the DONE bit of this port is set while the INTDONE bit of the UDC_TERM register is a 1 OR (2) when the RDYCHNG bit of this port is set while the INTRDCH bit of the UDC_TERM register is a 1. The INTPEND bit is cleared to 0 after any processor read of the DKC_STAT port. This also returns the controller chip's interrupt signal to its inactive state so that the next setting of the INTPEND bit generates another interrupt request.
DMAREQ	DMA request (bit 6). This bit is set to 1 whenever the controller chip requires a data transfer either to or from its data register UDC_DATA. This bit is cleared by such a data transfer.
DONE	Command done (bit 5). This bit is set to 1 when a command is complete. It is cleared to 0 (after a delay of 16 times the data bit transfer time) when a new command is issued. Note that the length of the delay depends upon the drive type and data rate options currently effective in the controller. The maximum time is 64 microseconds, which occurs when the controller is set up for a diskette drive with a data rate of 250 KHz (the slowest device).
TERMCOD	Termination code (bits 4:3). These bits indicate the conditions under which the most recent command terminated. They are valid only while the DONE bit of this port is set.

Bit 4	Bit 3	Condition
0	0	Successful completion
0	1	Error in READ ID sequence
1	0	Error in VERIFY sequence
1	1	Error in DATA TRANSFER sequence

Data Bit	Definition
	NOTE: *The following circumstances also result in a TERMCOD value of 11: the READY bit in the UDCDSTAT register is 0 at the completion of a DRIVE SELECT command, and the READY bit in the UDCDSTAT register is 1 at the completion of a DESELECT DRIVE command.*
RDYCHNG	Ready change (bit 2). This bit is set to 1 whenever the READY bit of the drive status register UDC_DSTAT changes state, either from 0 to 1 or vice versa. The RDYCHNG bit is cleared to 0 after any processor read of the DKC_STAT port.
	NOTE: *When a DRIVE SELECT or DESELECT DRIVE command is issued, or when the state of the INVRDY bit in register UDC_RTCNT is changed, the controller may detect a change in its ready input and set the RDYCHNG bit.*
OVRUN	Overrun/underrun (bit 1). This bit is set to 1 during a read or write command when the controller chip does not receive an acknowledgement of its DMA request in time to prevent loss of incoming data or a break in outgoing data. This bit is cleared to 0 by a RESET command to the controller or by a power-on.
BADSECT	Bad sector (bit 0). This bit is set to 1 when a bad sector (as indicated by the most significant bit of the head ID byte in the sector's ID field) is encountered. This bit is cleared when a new command is issued or a good sector is read.

NOTE: *As noted earlier, when the processor reads the DKC_STAT port, the port's INTPEND bit is cleared. If a device driver program sets up the disk controller to generate an interrupt request when DONE or RDYCHNG is set, then the program must not poll the DKC_STAT port while awaiting the interrupt. If the port is polled very close to the time that the DONE or RDYCHNG condition occurs, the controller chip may fail to signal the interrupt. Also, when a data transfer command (e.g. any of the READ, WRITE or FORMAT commands) has been issued to the controller, a program must not attempt to poll the DKCSTAT port while awaiting completion of the command, since the controller's path to the disk data buffer is also used by the CPU when it reads the chip controller ports. A program which wishes to use the controller in polled rather than interrupt mode should read bit DC in the INT_REQ register to monitor the state of the INTPEND bit of the DKC_STAT register.*

3.8.6.2 Controller Chip Registers

The controller chip contains fifteen 8-bit registers whose contents are accessible to the CPU via the controller chip ports, as described above. Table 3-27 shows the address of each register by a number in the range 0..A hex.

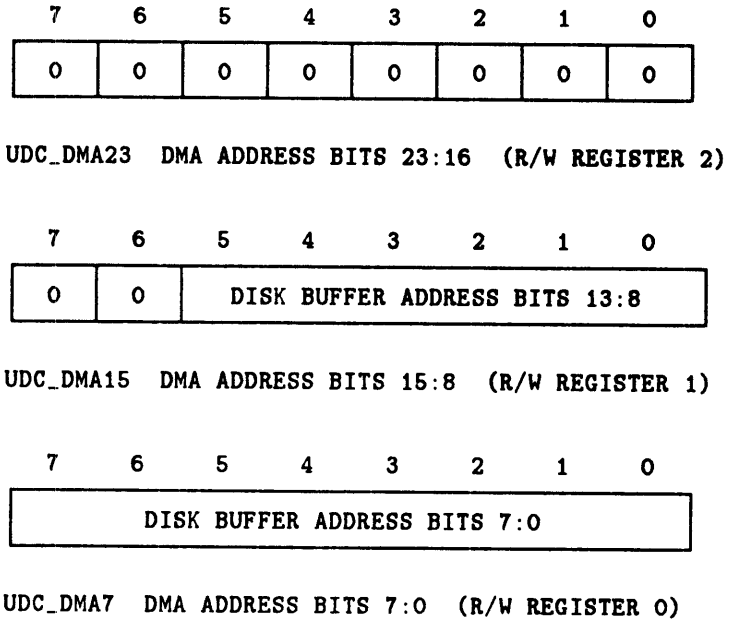
Table 3-27: Disk Controller Register Numbers

Number	Access	Name
0	r/w	UDC_DMA7 DMA address bits 7:0
1	r/w	UDC_DMA15 DMA address bits 15:8
2	r/w	UDC_DMA23 DMA address bits 23:16
3	r/w	UDC_DSECT Desired sector
4	wo	UDC_DHEAD Desired head
4	ro	UDC_CHEAD Current head
5	wo	UDC_DCYL Desired cylinder
5	ro	UDC_CCYL Current cylinder
6	wo	UDC_SCNT Sector count
6	ro[1]	(temporary storage)
7	wo	UDC_RTCNT Retry count
7	ro[1]	(temporary storage)
8	wo	UDC_MODE Operating mode
8	ro	UDC_CSTAT Chip status
9	wo	UDC_TERM Termination conditions
9	ro	UDC_DSTAT Drive status
A	r/w	UDC_DATA Data

3.8.6.2.1 DMA Address Registers (UDC_DMAxx)

The three 8-bit read/write DMA address registers form a 24-bit number which is used to address the disk data buffer during the data transfer portion of read and write commands. Since the buffer size is 16K bytes, only bits 13:0 of the DMA address are significant; bits 23:14 have no effect and should always be 0. Figure 3-73 shows the DMA address registers.

Figure 3-73: DMA Address Registers (UDC_DMAxx)

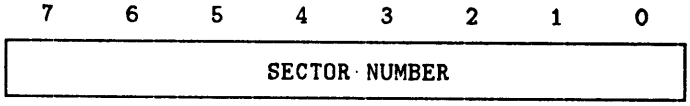


During multiple-sector read/write operations (except during the READ TRACK command), the DMA address contained in the UDC_DMAxx registers is incremented by the size of the sector after each successful read or write of a sector.

3.8.6.2.2 Desired Sector Register (UDC_DSECT)

The desired sector register (read/write register 3) is loaded with the starting sector number for each multiple-sector read/write operation (see Figure 3-74).

Figure 3-74: Desired Sector Register (UDC_DSECT)



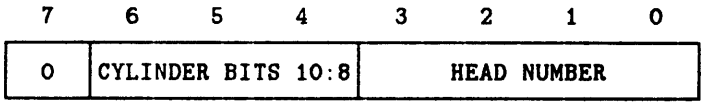
Except for the last sector of the operation, this register is incremented after each sector is successfully read or written. If the controller terminates a command because of an error in a sector, this register normally contains the number of the bad sector.

The range of valid sector numbers depends upon the drive type and the format of the medium in it. The nominal ranges are 0..16 for a hard disk, 1..10 for an RX50K diskette, and 1..15 for an RX33K diskette. However, the controller accepts any value in the range 0..255.

3.8.6.2.3 Desired Head Register (UDC_DHEAD)

The desired head register (write-only register 4) is loaded with the head number and the high-order bits of the cylinder number for the next command (see Figure 3-75).

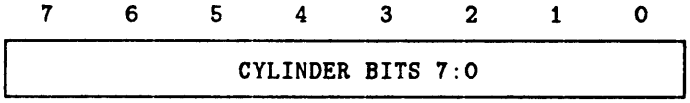
Figure 3-75: Desired Head Register (UDC_DHEAD)



3.8.6.2.4 Desired Cylinder Register (UDC_DCYL)

The desired cylinder register (write-only register 5) is loaded with the low-order bits of the cylinder number for the next command (see Figure 3-76).

Figure 3-76: Desired Cylinder Register (UDC_DCYL)



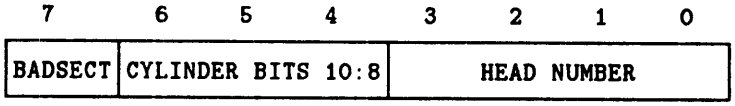
The UDC_DCYL and UDC_DHEAD registers specify the cylinder number and head number at which the next command is to begin. The range of valid values depends upon the selected drive.

CAUTION: *Be sure not to load a cylinder number larger than the number of physical cylinders in the selected drive. Attempting to exceed the existing number of cylinders may damage the drive.*

3.8.6.2.5 Current Head Register (UDC_CHEAD)

The current head register (read-only register 4) is loaded with the second byte of an ID field when a valid ID field sync mark is found during execution of a READ ID command sequence (see Figure 3-77).

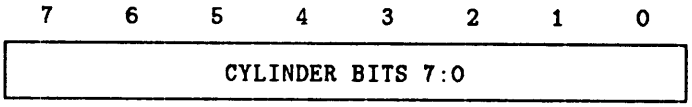
Figure 3-77: Current Head Register (UDC_CHEAD)



3.8.6.2.6 Current Cylinder Register (UDC_CCYL)

The current cylinder register (read-only register 5) is loaded with the first byte of an ID field when a valid ID field sync mark is found during execution of a READ ID command sequence (see Figure 3-78).

Figure 3-78: Current Cylinder Register (UDC_CCYL)

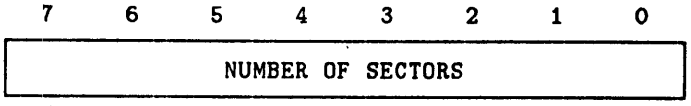


The UDC_CCYL and UDC_CHEAD registers return data from a disk ID field when a Read ID Field command sequence is executed as part of a command.

3.8.6.2.7 Sector Count Register (UDC_SCNT)

The sector count register (write-only register 6) is loaded with the number of sectors to be operated upon by a read or write command. An initial value of 0 results in an effective count value of 256. Figure 3-79 shows the sector count register.

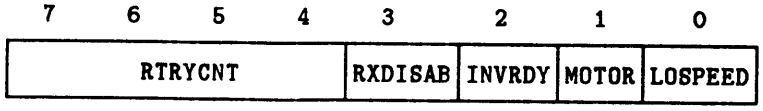
Figure 3-79: Sector Count Register (UDC_SCNT)



3.8.6.2.8 Retry Count Register (UDC_RTCNT)

The retry count register (write-only register 7) is loaded with the number of times the controller should retry a data field read operation before reporting an error. It also sets the state of four control signals (see Figure 3-80).

Figure 3-80: Retry Count Register (UDC_RTCNT)



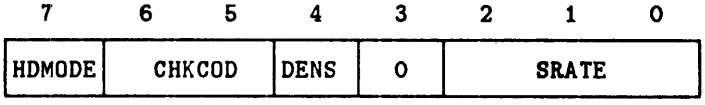
Data Bit	Definition
RTRYCNT	Retry count, in 1's complement form. For example, a value of 0 must be loaded as its complement, 1111. A non-0 value may be used only for READ LOGICAL commands; 0 must be used for all others.
RXDISAB	Disable diskette (bit 3). This bit determines whether the diskette drive is connected to the disk controller or is disconnected to allow an alternate controller to use the diskette drive. RXDISAB must be 0 for normal operation. When RXDISAB is 1, the diskette drive is entirely disconnected from the disk controller and cannot be used until RXDISAB is set to 0 again. This bit is set to 0 (enabled) by power-on and by an IORESET. This bit does not affect the operation of hard disk drives.
INVRDY	Invert ready (bit 2). This bit determines the polarity of the diskette drive's status signal which is interpreted as "drive ready" by the controller chip and which appears as a 1 in the READY bit of the UDC_DSTAT register. When INVRDY is 0, a "low" status signal from the diskette drive asserts the "drive ready" condition and appears as a 1 in the READY bit. When INVRDY is 1, a "high" status signal from the diskette drive asserts the "drive ready" condition to the controller chip and appears as a 1 in the READY bit of UDC_DSTAT. When the diskette drive is selected, INVRDY must be used as described in Section 3.8.10 to cause the RX33 drive's status signal to be seen as "drive ready" by the controller. Hard disk drives are not affected by INVRDY. However, for compatibility with early systems, INVRDY should be set to 0 when a hard disk is selected.
MOTOR	Motor on (bit 1). When this bit is set to 1, the motor of the diskette drive is turned on. The state of this bit has no effect on hard disk drives.

Data Bit	Definition
LOSPEED	Diskette speed select (bit 0). This bit selects the rotation speed and data rate of RX33 diskette drives. When it is 0, the speed is 360 rpm and the data rate is 500 KHz (required for RX33K media). When it is 1, the speed is 300 rpm and the data rate is 250 KHz (required for RX50K media). The state of this bit has no effect on hard disk drives. **NOTE:** *The settings of the RXDISAB, INVRDY, MOTOR, and the LOSPEED bits are transmitted to the hardware only when a DRIVE SELECT or DESELECT DRIVE command is issued. Loading new values into UDC_RTCNT does not by itself have any effect; one of those two commands must subsequently be issued to make the bits effective. A reset caused by power-on or a write to the IORESET register clears these four 4 bits to 0 and immediately transmits those values to the hardware (thus the diskette will be connected to the disk controller and its drive motor will stop).*

3.8.6.2.9 Operating Mode Register (UDC_MODE)

The operating mode register (write-only register 8) sets the operating mode of the controller to accomodate various drive types (see Figure 3–81). Table 3–28 lists the mode values for the drives supported.

Figure 3–81: Operating Mode Register (UDC_MODE)



Data Bit	Definition
HDMODE	Hard disk mode (bit 7). This bit controls whether the controller read data input is to be level transitions or pulse inputs. For this system, this bit must be 1 for both hard disk and diskette drives.
CHKCOD	Error checking code (bits 6:5). These bits select the error checking code which is generated during writing and checked during reading.

6	5	Error Checking Code
0	0	CRC code. This is to be used for all types of diskettes.
1	0	Internal 32-bit ECC without automatic correction. This is to be used with hard disks (correction under software control).

DENS	Density select (bit 4). When this bit is 1, data is recorded in single-density FM mode. When this bit is 0, data is recorded in double-density MFM mode. This bit should always be 0 for both diskettes and hard disks.
	Bit 3 is not used and must be 0.
SRATE	Seek step rate (bits 2:0). These bits set the rate at which cylinder step pulses are issued by the controller during seek operations. The rate is also affected by the type of drive (bit HDMODE in this register and bits 3:2 of the most recent DRIVE SELECT command), and by the recording density (bit DENS in this register).

Data Bit	Definition		
	2	1	0	Cylinder Step Pulse Rates
	0	0	1	RX33 diskette drive operated at 300 rpm/250 KHz (required for RX50K and 48 tpi media). Step period is 4 milliseconds.
	0	1	0	RX33 diskette drive operated at 360 rpm/500 KHz (required for RX33K media). Step period is 4 milliseconds.
	0	0	0	Normal commands to hard disk drives. Step period is 17.6 microseconds.
	1	1	0	RESTORE DRIVE commands to all hard disk drives. Step period is 6.4 milliseconds.

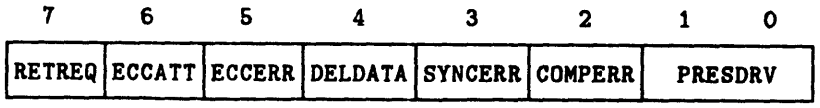
Table 3-28: Mode Values for the Drives

Drive and Media	HDMODE	CHKCOD	DENS	0	SRATE
RX33 drive with RX50K media	1	00	0	0	001
RX33 drive with 48tpi media	1	00	0	0	001
RX33 drive with RX33K media	1	00	0	0	010
RDxx hard disk (normal)	1	10	0	0	000
RDxx hard disk (RESTORE)	1	10	0	0	110

3.8.6.2.10 Chip Status Register (UDC_CSTAT)

The chip status register (read-only register 8) supplies additional chip status information. The contents of this register are valid only between the time that the DONE bit in the interrupt status port DKC_STAT is set and the time the next command is written to the controller command port DKC_CMD. Figure 3-82 shows the chip status register.

Figure 3-82: Chip Status Register (UDC_CSTAT)



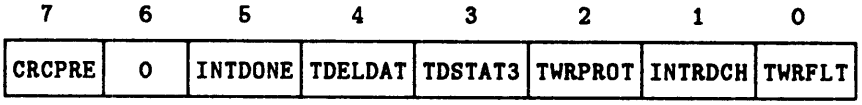
Data Bit	Definition
RETREQ	Retry required (bit 7). This bit is set to 1 if a retry was attempted by the controller during the execution of any read command.
ECCATT	Error correction attempted (bit 6). This bit is set to 1 if the controller's internal ECC logic has attempted to correct a bad sector.
ECCERR	ECC/CRC Error (bit 5). This bit is set to 1 if the controller detects a CRC or ECC error while reading from a disk.
DELDATA	Deleted data mark (bit 4). This bit is set to 1 when the controller reads a sector ID field which has a "deleted data" mark. This bit is set to 0 for normal sector ID fields.
SYNCERR	Synchronization error (bit 3). This bit is set to 1 if the controller does not find a sync mark while it is attempting to read either an ID or a data field. The command being executed is terminated when this bit is set.
COMPERR	Compare error (bit 2). The bit is set to 1 if the information contained in the desired cylinder and desired head registers (UDC_DCYL and UDC_DHEAD) does not match that in an ID field read from a disk. The command being executed is terminated when this bit is set.
PRESDRV	Present drive selected (bits 1:0). These bits represent the number of the drive currently selected by the controller.

1	0	Drive Selected
0	0	First hard disk drive
0	1	Second hard disk drive
1	0	Diskette drive

3.8.6.2.11 Termination Conditions Register (UDC_TERM)

The termination conditions register (write-only register 9) selects the conditions which terminate a command and those which generate an interrupt request to the processor (see Figure 3-83).

Figure 3-83: Termination Conditions Register (UDC_TERM)



Data Bit	Definition
CRCPRE	CRC register preset (bit 7). When this bit is set to 1, the CRC/ECC registers are preset to 1 for error code generation and checking. A value of 1 is required for both diskettes and hard disks.
	Bit 6 is not used and must be 0.
INTDONE	Interrupt on done (bit 5). When this bit is set to 1, the setting of the command completion bit DONE in the interrupt status port DKC_STAT will also set the INTPEND bit in that port and signal a hardware interrupt request to the system interrupt controller. When INTDONE is 0, INTPEND is not set and no interrupt request is signalled.
TDELDAT	Terminate on deleted data (bit 4). While this bit is set to 1, if the DELDATA bit in the chip status register UDC_CSTAT is set by the detection of a deleted data mark in a sector ID field, the current command terminates (and the DONE bit in DKC_STAT is set) when the current sector operation is completed.
TDSTAT3	Terminate on drive status 3 change (bit 3). While this bit is set to 1, if the DSTAT3 bit in the drive status register UDC_DSTAT is set to 1, the current command terminates (and the DONE bit in DKC_STAT is set) when the current sector operation is completed.
TWRPROT	Terminate on write protect (bit 2). While this bit is set to 1, if the WRPROT bit in the drive status register UDC_DSTAT is set by a write protect signal from the selected drive, the current WRITE or FORMAT TRACK command terminates (and the DONE bit in DKC_STAT is set).
	NOTE: *Write protect can be signalled only by a diskette drive.*

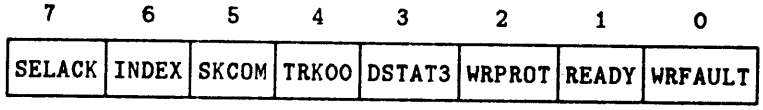
Data Bit	Definition
INTRDCH	Interrupt on ready change (bit 1). When this bit is set to 1, the setting of the ready change bit RDYCHNG in the interrupt status port DKC_STAT also sets the INTPEND bit in that port and signal a hardware interrupt request to the system interrupt controller. When INTRDCH is 0, INTPEND is not set and no interrupt request is signalled.
TWRFLT	Terminate on write fault (bit 0). While this bit is set to 1, if the WRFAULT bit in the drive status register (UDC_DSTAT) is set by a write fault signal from the selected drive, the current WRITE or FORMAT TRACK command terminates (and the DONE bit in DKC_STAT is set) when the current sector operation is completed.
	NOTE: *Write fault can be signalled only by a hard disk drive.*

NOTE: *The contents of the UDC_TERM register are destroyed whenever a RESET command is issued or an I/O reset signal is received. In particular, INTDONE is cleared so that the chip does not generate any command done interrupts until UDC_TERM is set up again.*

3.8.6.2.12 Drive Status Register (UDC_DSTAT)

The drive status register (read-only register 9) shows the state of several signals from the currently selected drive. Its contents are invalid if no drive is selected. Figure 3-84 shows the drive status register.

Figure 3-84: Drive Status Register (UDC_DSTAT)



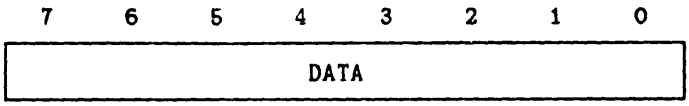
Data Bit	Definition
SELACK	Select acknowledge (bit 7). This bit is 1 when a select acknowledge signal is received from the currently selected hard disk drive. Failure to receive this signal indicates that no drive is installed to respond to the current drive select number. SELACK is always 0 for diskette drives.
INDEX	Index point (bit 6). This bit is 1 when the current drive's medium passes its index point. The duration of the 1 state varies depending upon the drive type and (for diskettes) the speed selected.
SKCOM	Seek complete (bit 5). This bit is 0 while the currently selected hard disk drive is moving its heads; it becomes 1 when the drive has completed the seek operation and its heads are stable. SKCOM is always 1 when a diskette drive is selected; it cannot be used to delay for seek settling time for a diskette drive (such a delay must be provided by the driver software).
TRK00	Track 0 (bit 4). This bit is 1 when the currently selected drive's heads are positioned at cylinder 0. It is valid for all drive types.
DSTAT3	Drive status 3 (bit 3). This bit is unused and is always set to 0.
WRPROT	Write protect (bit 2). This bit reflects the state of the write protect signal received from the currently selected drive: a 1 indicates that writing is prohibited. For a diskette drive, WRPROT is 1 when the diskette in the drive has its write-protect notch covered. For a hard disk drive, this bit is always 0.
READY	Drive ready (bit 1). This bit indicates whether or not the controller chip perceives that the currently selected drive is ready for operation. When READY is 1, the controller issues head positioning and data transfer commands to the drive. When READY is 0, the controller does not execute such commands. The state of READY for the diskette drive is determined by the drive status signal from the currently selected drive and the setting of the INVRDY bit of the UDC_RTCNT register. When INVRDY is 0, a "low" drive status signal makes READY a 1; when INVRDY is 1, a "high" drive status signal makes READY a 1. INVRDY must be used as described in Section 3.8.10 to make READY a 1 so that the controller issues commands to the drive. For hard disk drives, READY is always 1 when the drive is ready for operation and INVRDY does not affect the READY polarity. However, for compatibility with early systems, INVRDY should always be 0 for hard disk drives.

Data Bit	Definition
WRFAULT	Write fault (bit 0). This bit is 1 when the selected hard disk drive finds an internal condition which prevents successful write operations, such as improper supply voltages. This bit is always 0 for diskettes.

3.8.6.2.13 Disk Data Register (UDC_DATA)

The disk data register (read/write register 0Ah) is used by the controller's DMA logic to pass data to and from the disk during data transfer operations. It is also used by a program to specify the head load time delay for a DRIVE SELECT command. Figure 3-85 shows the disk data register.

Figure 3-85: Disk Data Register (UDC_DATA)



NOTE: *The controller chip internal register pointer must be set to 0Ah by a SET REGISTER POINTER command to designate the UDC_DATA register during all DMA data transfer operations.*

3.8.7 Command Overview

The controller executes fourteen commands, which can be divided into two groups. The first group comprises housekeeping and control operations which do not transfer data to or from a drive.

- RESET
- SET REGISTER POINTER
- DESELECT DRIVE
- DRIVE SELECT
- RESTORE DRIVE
- STEP
- POLL DRIVES

The second group of commands transfer data to or from a drive.

- SEEK/READ ID
- FORMAT TRACK

- READ TRACK
- READ PHYSICAL
- READ LOGICAL
- WRITE PHYSICAL
- WRITE LOGICAL

The controller has an internal status byte which it checks at various times during command execution. This byte contains copies of the DELDATA bit (in the UDC_CSTAT register), the BADSECT and OVRUN bits (in the DKC_STAT port), and the READY, WRPROT, WRFAULT, and CARTCH bits (in the UDC_DSTAT register). This internal status byte is examined before the execution of all READ and WRITE commands and is checked again just prior to the completion of most commands. It is also checked between sector operations during the execution of READ LOGICAL, READ PHYSICAL, WRITE LOGICAL, and WRITE PHYSICAL commands. The controller makes decisions regarding command termination and interrupt generation based upon the contents of this status byte and the state of the bits in the UDC_TERM register.

At the completion of all commands, the controller sets the DONE bit in the DKC_STAT port. Depending upon the contents of the UDC_TERM register, this may also generate an interrupt request, except for the RESET and SET REGISTER POINTER commands which never generate interrupt requests. Issuing a new command clears the DONE bit.

During all data transfer commands (except READ TRACK), the controller uses three common sequences of internal operations. As it begins each sequence, the controller places a code identifying it in the TERMCOD bits of the DKC_STAT port. If the command is not completed successfully, these bits identify the sequence during which the failure occurred. The sequences and codes are:

0 1 READ ID

1 0 VERIFY

1 1 DATA TRANSFER.

3.8.7.1 Read ID Sequence

The READ ID sequence reads the next available ID field (using the head designated by the UDC_CHEAD register) to find the cylinder at which the heads are positioned and then, if necessary, moves the heads to the position specified in the desired cylinder registers UDC_DCYL and UDC_DHEAD. The sequence comprises the following steps:

1. Attempt to find an ID field sync mark. If no mark is found within 33,792 byte times, the controller sets the SYNCERR bit of the UDC_CSTAT register and terminates the command.
2. Read the ID field. The data from the ID field is stored in the UDC_CCYL and UDC_CHEAD registers. If the CRC bytes of the ID field are incorrect, the controller sets the ECCERR bit of the UDC_CSTAT register and terminates the command.
3. Move to desired cylinder. The controller calculates the direction and number of step pulses required to move the heads from their current position to that specified in the UDC_DCYL and UDC_DHEAD registers, and (if necessary) issues the step pulses to the drive.

3.8.7.2 Verify Sequence

The VERIFY sequence reads ID fields on the current track to verify that the heads are at the desired cylinder, that the head number is correct, and to find the desired sector for a data transfer. The sequence comprises the following steps:

1. Attempt to find an ID field sync mark. If no mark is found within 33,792 byte times, the controller sets the SYNCERR bit of the UDC_CSTAT register and terminates the command.
2. Search for desired sector. The data from the ID field is compared with the contents of the UDC_DCYL, UDC_DHEAD, and UDC_DSECT registers. If the contents match, the sequence continues with step 3. Otherwise, the controller hunts for the next ID field sync mark and repeats the comparison process. If the desired sector is not found within 33,792 byte times, then the COMPERR bit in the UDC_CSTAT register is set and the command is terminated.
3. Check the ID field validity. When the desired sector is found, if the CRC bytes of the ID field are incorrect, the controller sets the ECCERR bit of the UDC_CSTAT register and terminates the command.

   For READ PHYSICAL and WRITE PHYSICAL commands, the ID field comparison is done only until the first sector to be transferred is found. For subsequent sectors, the ID field contents are not compared, although the ID field CRC is checked.

3.8.7.3 Data Transfer Sequence

The DATA TRANSFER sequence transfers the contents of the next available data field to or from the disk data buffer. For a READ operation, the sequence comprises the following steps:

1. Find data sync mark. The controller searches for a data sync mark (FBh or F8h). If the mark is F8h, then the controller sets the DEL-DATA bit in the UDC_CSTAT register; otherwise it clears that bit. When the data sync mark is found, the controller updates the UDC_CCYL and UDC_CHEAD registers from the values found in the ID field preceding the data sync mark.

2. Perform DMA transfer. Using DMA, the controller transfers the data bytes and the CRC/ECC bytes of the sector to the disk data buffer. If the system does not respond to DMA requests from the controller within 1 byte time, the controller sets the OVRUN bit in the DKC_STAT port and terminates the command.

3. Check CRC/ECC bytes. If the CRC/ECC bytes following the data are incorrect and the controller cannot correct the data (or has been instructed not to try, according to the CHKCOD bits of the UDC_MODE register), then the controller sets the RETREQ bit in the UDC_CSTAT register and decrements the RTRYCNT field of the UDC_RTCNT register. If the UDC_RTCNT register is now 0, then the controller sets the ECCERR bit in the UDC_CSTAT register and terminates the command. Otherwise, the controller goes back to the VERIFY sequence to locate the sector for another attempt.

For a WRITE operation, the sequence comprises the following steps:

1. Write data sync mark. The controller writes either a normal or deleted data mark according to the write command byte.

2. Perform DMA transfer. Using DMA, the controller transfers the data bytes from the disk data buffer to the sector. If the system does not respond to DMA requests from the controller within 1 byte time, the controller sets the OVRUN bit in the DKC_STAT port and terminates the command.

3. Write CRC/ECC bytes. The controller writes the CRC/ECC bytes following the data. Note that no error retries are permitted for write operations, so the RTRYCNT field of the UDC_RTCNT register should be set to 0 (1s complement form).

After each successful sector transfer, the controller adds the size of the sector data (not including its CRC/ECC bytes) to the UDC_DMAx registers and decrements the UDC_SCNT register. If the UDC_SCNT register is then 0, the controller terminates the command. Otherwise, the controller increments the UDC_DSECT register, resets the RTRYCNT field of the UDC_RTCNT register to its value as of the beginning of the command, and returns to the VERIFY sequence to locate the next sector.

When the controller reads a sector, it transfers the sector's error checking bytes (2 bytes for diskette CRC; 4 bytes for hard disk ECC) into the disk data buffer following the sector's last data byte. If the read is successful, the DMA address is advanced only by the number of data bytes, so the data from the next sector of a multi-sector read will be contiguous with the preceding sector's data. However, the buffer must have space to hold the error checking bytes of the last sector, so the highest allowable starting point in the buffer is the buffer size (16384) minus the sector size (512 + 2 or 512 + 4 bytes). If the DMA address exceeds the buffer size, it wraps around to the beginning of the buffer.

3.8.8 Command Descriptions

This section describes the disk commands.

3.8.8.1 RESET Command

The RESET command places the controller chip in a known state. It has the same effect as a power-on reset. The DONE bit in the DKC_STAT port is set by this command but no interrupt request is generated. This is because execution of this command clears the UDC_TERM register. The UDC_TERM register must be reloaded after executing this command. A program may issue a RESET command to terminate the execution of any non-data-transfer command, but data transfer commands cannot be terminated in this manner. Figure 3-86 shows the RESET command.

Figure 3-86: RESET Command

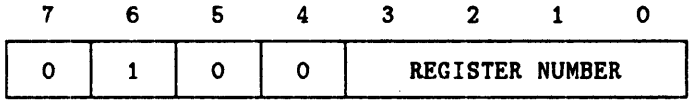
7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0

3.8.8.2 SET REGISTER POINTER Command

The SET REGISTER POINTER command sets the controller's internal register pointer to designate the register which is accessed by the next CPU access to the DKC_REG port. Note that each such CPU access increments the internal pointer until it reaches its highest value of 0Ah (register UDC_DATA), after which the pointer remains at this value.

Do not set the pointer to a value outside the valid range of 00h through 0Ah. The DONE bit in the DKC_STAT port is set by this command but no interrupt request is generated. Figure 3-87 shows the SET REGISTER POINTER command.

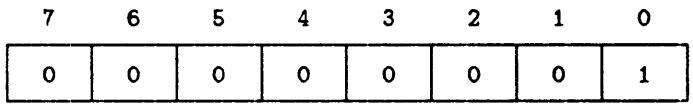
Figure 3-87: SET REGISTER POINTER Command



3.8.8.3 DESELECT DRIVE Command

The DESELECT DRIVE command negates all drive select outputs so that no drive is selected. When no drive is selected, the contents of the drive status register UDC_DSTAT are invalid. Figure 3-88 shows the DESELECT DRIVE command.

Figure 3-88: DESELECT DRIVE Command



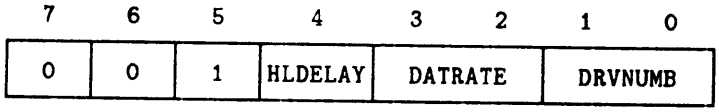
NOTE: *The DESELECT DRIVE command should be issued when no drive is in use. Execution of this command may cause RDYCHNG to be set in the DKC_STAT port.*

Execution of this command transmits the values of the INVRDY, MOTOR, RXDISAB, and LOSPEED bits from the UDC_RTCNT register to the hardware. If the READY bit of the UDC_DSTAT register is 1 at the conclusion of this command because the INVRDY bit of the UDC_RTCNT register was 1 at the time the DESELECT DRIVE command was issued, the TERMCOD bits of the DKC_STAT register will be 11. This does not indicate an error and should be ignored.

3.8.8.4 DRIVE SELECT Command

The DRIVE SELECT command selects 1 of the four possible drives connected to the controller and sets its data transfer rate (see Figure 3-89).

Figure 3-89: DRIVE SELECT Command



Data Bit	Definition
HLDELAY	Head load delay (bit 4). When this bit is set, the controller delays for diskette head loading at the beginning of data transfer commands. The duration of the delay is specified by the contents of the UDC DATA register at the time that the command is issued. The RX33 drives do not require this delay; this bit should be 0 for all diskette and hard disk drives.
DATRATE	Data rate (bits 3:2). These bits determine the data bit rate and hard disk format options.

Bit 3	Bit 2	Data Rate
0	0	Hard disk with 3-byte ID fields. Not used in this system.
0	1	Hard disk with 4-byte ID fields. Use this value for all hard disks.
1	0	Diskette with 500 KHz data rate. Use this value for diskette drives with RX33K high-capacity media.
1	1	Diskette with 250 KHz data rate. Use this value for diskette drives with standard media, including RX50K and 48 tpi media.

DRVNUMB Drive number (bits 1:0). These bits select the active drive.

Data Bit	Definition

1	0	Drive Selected
0	0	First hard disk (in VS410 system unit)
0	1	Second hard disk (in VS40B storage expansion unit)
1	0	Diskette drive (in VS410 system unit)

The DRIVE SELECT command transfers the contents of the desired head register UDC_DHEAD to the current head register UDC_CHEAD. When any command which uses a READ ID sequence (for example, a read or write command) is executed, the head designated by the UDC_CHEAD register is used to find the present position on the disk. This requires that UDC_CHEAD designate a head which is valid for the selected drive and medium. Therefore, prior to issuing a DRIVE SELECT command, a program must load the UDC_DHEAD register with a head number (in bits 3:0) which is valid for both the drive and medium being selected (head 0 is the best choice, since it's guaranteed to be valid for any case).

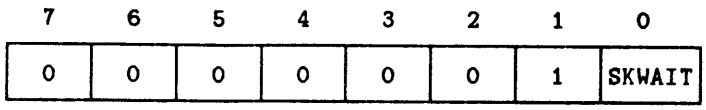
Execution of DRIVE SELECT transmits the values of the INVRDY, MOTOR, RXDISAB, and LOSPEED bits from the UDC_RTCNT register to the hardware. If the READY bit of the UDC_DSTAT is 0 at the conclusion of this command (this depends upon the drive's status signal and the value of the INVRDY bit), the TERMCOD bits of the DCK_STAT register will be 11. If the selected drive is a hard disk, this is a "not ready" error condition. For a diskette drive, this may not be an error. Section 3.8.10 explains READY and INVRDY for diskettes.

Whenever a DRIVE SELECT command selects a diskette drive that was not already selected, up to 70 milliseconds may be required for the read data recovery circuit to stabilize before the controller receives usable data from the diskette. No delay is required when selecting a hard disk drive.

3.8.8.5 RESTORE DRIVE Command

The RESTORE DRIVE command sends step pulses to the selected drive to move its heads outward until it reaches cylinder 0. Prior to issuing this command, a drive must have been selected by a DRIVE SELECT command and the UDC_MODE register must be set for the selected drive type. Figure 3-90 shows the RESTORE DRIVE command.

Figure 3-90: Restore Drive Command

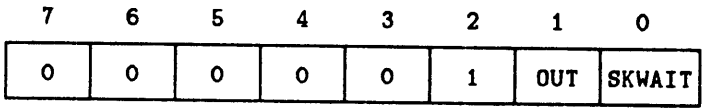


Data Bit	Definition
SKWAIT	Wait for seek complete (bit 0). If this bit is 1, the controller tests the seek complete signal from the drive (reflected in the SKCOM bit in the UDC_DSTAT register) to determine when head motion is complete. If SKWAIT is 0, the controller assumes that motion is complete after it has issued the last step pulse. SKWAIT should be 0 for diskette drives and 1 for hard disk drives. Before issuing each step pulse, the controller checks the TRK00 and READY bits in the UDC_DSTAT register. If TRK00 is 1 or READY is 0, the controller terminates the command. The controller issues up to 4096 step pulses, checking TRK00 and READY after each one. If the drive does not set TRK00 to 1 during this time, then the controller terminates the command with the TERMCOD bits in the DKC_STAT port set to 10. This command requires that the READY bit in the UDC_DSTAT register be 1. Section 3.8.10 explains the READY state for diskette drives. **NOTE:** *When attempting to RESTORE a hard disk, be sure to set the step rate to 6.4 milliseconds for non-buffered seeks.*

3.8.8.6 STEP Command

The STEP command issues one step pulse to move the heads of the selected drive in or out 1 cylinder. Prior to issuing this command, a drive must have been selected by a DRIVE SELECT command and the UDC_MODE register must be set for the selected drive type. Figure 3-91 shows the STEP command.

Figure 3-91: STEP Command



Data Bit	Definition
OUT	Direction of motion (bit 1). If it is 1, motion is outward toward cylinder 0. If it is 0, motion is inward. Care must be taken not to attempt to move the heads inward beyond the number of cylinders on the device.
SKWAIT	Wait for seek complete (bit 0). If this bit is 1, the controller tests the seek complete signal from the drive (reflected in the SKCOM bit in the UDC_DSTAT register) to determine when head motion is complete. If SKWAIT is 0, the controller assumes that motion is complete after it has issued the last step pulse. SKWAIT should be 0 for diskette drives and 1 for hard disk drives.

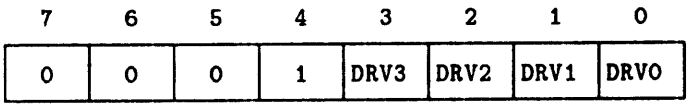
The STEP command is normally used during formatting. It does not attempt to read an ID field to verify its position and so it works on an unformatted disk.

This command requires that the READY bit in the UDC_DSTAT register be 1. Section 3.8.10 explains the READY state for diskette drives.

3.8.8.7 POLL DRIVES Command

The POLL DRIVES command polls selected drives for seek complete signals to assist a driver program to perform simultaneous seeks on hard disk drives (see Figure 3-92).

Figure 3-92: POLL DRIVES Command



Data Bit	Definition
DRVx	Drives to be polled (bits 3:0). These bits determine which drives are polled. A 1 includes a drive in the poll sequence. Since only hard disk drives delay assertion of the seek complete signal until their head motion is complete, only bits DRV0 and DRV1 should ever be set. Seek complete is asserted at once, whenever a diskette drive is selected. The command operates by selecting in turn each drive whose DRVx bit was set in the POLL DRIVES command until a drive is polled whose seek complete signal is set (this signal appears in bit SKCOM in the UDC_DSTAT register), at which point the controller terminates the command. At the completion of the command, the PRESDRV bits of the UDC_CSTAT register indicate which drive is selected. The driver program must explicitly select each drive from which it expected a seek complete signal and test its value in the SKCOM bit of the UDC_DSTAT register.

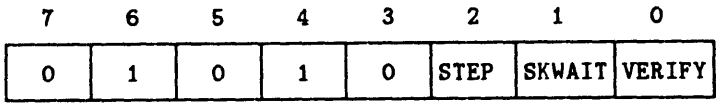
The POLL DRIVES command must be preceded by a DESELECT DRIVE command.

3.8.8.8 SEEK/READ ID Command

The SEEK/READ ID command determines where the heads of the selected drive are presently positioned by performing a READ ID sequence. Then, if the STEP option bit in the command code is set, it moves the heads to the new position determined by the UDC_DCYL and UDC_DHEAD registers. Figure 3-93 shows the SEEK/READ ID command.

Prior to executing this command, a drive must have been selected by a DRIVE SELECT command and the following registers must be appropriately set; UDC_MODE, UDC_RTCNT, and (if STEP is set) UDC_DCYL and UDC_DHEAD.

Figure 3-93: SEEK/READ ID Command

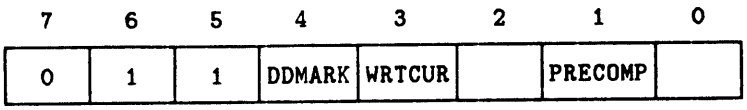


Data Bit	Definition
STEP	Seek to desired cylinder (bit 2). If this bit is 1, the controller issues the step pulses necessary to move the heads from their current position to that specified by the UDC_DCYL and UDC_DHEAD registers. If the STEP bit is 0, no motion occurs and the only effect of the command is to update the UDC_CCYL and UDC_CHEAD registers to reflect the current position of the heads.
SKWAIT	Wait for seek complete (bit 1). If this bit is 1, the controller tests the seek complete signal from the drive (reflected in the SKCOM bit in the UDC_DSTAT register) to determine when head motion is complete. If SKWAIT is 0, the controller assumes that motion is complete after it has issued the last step pulse. SKWAIT should be 0 for diskette drives and 1 for hard disk drives. This bit must be 0 if the STEP bit is also 0.
VERIFY	Verify position (bit 0). If this bit is 1, the controller performs a VERIFY sequence after performing the operations indicated by the STEP and SKWAIT bits. This bit must be 0 if the STEP bit is also 0.

3.8.8.9 FORMAT TRACK Command

The FORMAT TRACK command writes on the current track a complete new image consisting of sector ID fields and data fields with the appropriate gaps between them. It writes the entire track, beginning at the leading edge of the index signal and continuing until the index signal is received again. This command must be used to format each track of a disk before any other data transfer command can be issued to that disk. Note that this command does not perform READ ID and VERIFY sequences; it writes to the currently selected cylinder and head. Figure 3-94 shows the FORMAT TRACK command.

Figure 3-94: FORMAT TRACK Command



Data Bit	Definition
DDMARK	Deleted data mark (bit 4). If this bit is 1, each data field is preceded by a deleted data mark (F8h). Otherwise the data fields are preceded by a normal data mark (FBh).
WRTCUR	Reduced write current (bit 3). Not used and must be 0.
PRECOMP	Write precompensation. Section 3.8.9 explains write precompensation.

Prior to issuing this command, the controller must have selected the drive with a DRIVE SELECT command and positioned the heads to the correct cylinder using the RESTORE DRIVE and STEP commands. These commands can be used on an unformatted disk. The SEEK/READ ID command cannot be used on an unformatted disk because it attempts to read ID fields from the disk.

The information bytes for each sector's ID field are read from a table in the disk data buffer. This table must have four bytes for each sector to be established on the track. Figure 3-95 shows the contents of the table in the disk data buffer.

Figure 3-95: ID Field Bytes for Each Sector

	7	6	5	4	3	2	1	0
0	CYLINDER BITS 7:0							
1	BADSECT	CYLINDER BITS 10:8			HEAD NUMBER			
2	SECTOR NUMBER							
3	0	0	0	0	0	0	1	0

For diskettes: Byte 0 contains the track number 0..79; byte 1 contains the head number in bit 0 and is otherwise 0. Byte 2 contains the sector number in the range 1..15 for RX33K media. And byte 3 indicates a sector size of 512 data bytes.

For hard disks: Byte 0 and bits 6:4 of byte 1 contain the cylinder number. Byte 1 contains the head number and bad-sector flag. Byte 2 contains the sector number in the range 0..16. And byte 3 indicates a sector size of 512 data bytes followed by 4 ECC bytes. The order of sector numbers may be arranged to provide whatever interleave factor is desired.

The BADSECT bit in the second byte of the sector ID field is set to 1 to flag a physically defective sector. (The driver program must provide a means of substituting another sector for the defective 1.) There must be at least 1 sector on each track which is NOT marked with the BADSECT bit. If the controller chip encounters a track all of whose sectors have BADSECT set, the chip functions unpredictably.

The FORMAT TRACK command requires a large number of parameters, so some registers must be used twice. The following steps are required to format a track:

1. Set up the information for the ID field bytes in the disk data buffer and load the UDC_DMAx registers with the address of the information for the first sector. Then issue a DRIVE SELECT command to select the proper drive. An additional effect of this command is to save the contents of the UDC_DMAx registers in the UDC_CHEAD and UDC_CCYL registers and a temporary register so that the UDC_DMAx registers can be reused to supply additional format parameters.
2. Load the UDC_DHEAD register with the correct head number.

3. Load the parameters listed in Table 3-29 into the registers and in the formats indicated (note that RX50K and 48 tpi media cannot be formatted by this system). The values are listed in decimal true form, before conversion to the format required by their registers.

Table 3-29: Register Parameters

Parameter	Register	Format	Hard disk Value	RX33K value
Gap 0 size	UDC_DMA7	2's comp	16	80
Gap 1 size	UDC_DMA15	2's comp	16	50
Gap 2 size	UDC_DMA23	2's comp	5	22
Gap 3 size	UDC_DSECT	2's comp	40	84
Sync size	UDC_DCYL	1's comp	13	12
Sector count	UDC_SCNT	1's comp	17	15
Sector size code	UDC_RTCNT	1's comp	4	4

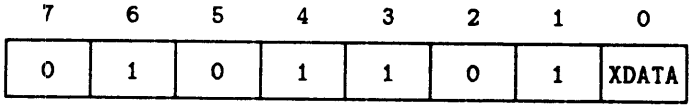
4. Load the UDC_MODE register as appropriate for the drive and medium.
5. Position to the desired cylinder, using RESTORE DRIVE or STEP commands.
6. Issue the FORMAT TRACK command. All data field bytes are filled with a value of E5h and all gaps are filled with 4Eh.

Additional tracks under the same head can be formatted by revising the ID field bytes in the disk data buffer and repeating steps 5 and 6. When it is necessary to select a new head, the entire sequence of steps must be repeated.

3.8.8.10 READ TRACK Command

The READ TRACK command reads the ID fields and (optionally) the data fields from an entire track into the disk data buffer, starting from the index point and ending when the index point is again reached. No error checking is performed on ID or data fields. Note that this command does not perform READ ID and VERIFY sequences; data is read from the currently selected cylinder and head. Figure 3-96 shows the READ TRACK command.

Figure 3-96: READ TRACK Command



Data Bit	Definition
XDATA	Transfer data fields (bit 0). If this bit is 1, data fields as well as ID fields are transferred from each sector into the disk data buffer. If it is 0, only ID fields are transferred.

Prior to executing this command, a drive must have been selected by a DRIVE SELECT command, the following registers must have been loaded, and the internal register pointer must be set to point to the UDC_DATA register.

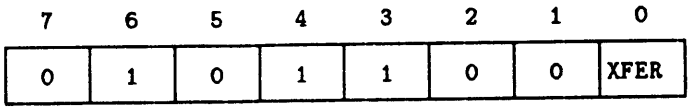
- UDC_MODE (mode appropriate to selected drive and media).
- UDC_DMAx (starting address in disk data buffer).
- UDC_RTCNT (retry count. The RTRYCNT field must be 0 (1s complement form) for this command. Automatic retries cannot be performed during this command).

NOTE: *Unlike the normal READ and WRITE commands, the READ TRACK command does not update the UDC_DMAx (zzz) registers to reflect the amount of data placed in the disk buffer.*

3.8.8.11 READ PHYSICAL Command

The READ PHYSICAL command reads 1 or more sectors from a track, beginning with a specified sector and continuing through physically consecutive sectors, until either the sector count is satisfied, or a bad sector is encountered, or the track index is reached. Figure 3-97 shows the READ PHYSICAL command.

Figure 3-97: READ PHYSICAL Command



Data Bit	Definition
XFER	Transfer data (bit 0). If this bit is 1, data is transferred from each sector into the disk data buffer. If it is 0, no data is transferred but all error checking is still performed.

Prior to executing this command, a drive must have been selected by a DRIVE SELECT command, the following registers must have been loaded, and the internal register pointer must be set to point to the UDC_DATA register:

- UDC_MODE (mode appropriate to selected drive and media).
- UDC_DMAx (starting address in disk data buffer).
- UDC_DCYL (desired cylinder).
- UDC_DHEAD (desired head).
- UDC_DSECT (number of first sector).
- UDC_SCNT (number of sectors to be read).
- UDC_RTCNT (retry count. The RTRYCNT field must be 0 (1s complement form) for this command. Automatic retries cannot be performed because the number of the sector to be retried (after the first one) is not necessarily the same as that in the UDC_DSECT register).

The controller begins command execution by using the READ ID, VERIFY, and DATA TRANSFER sequences to find and read the first sector. After this and each subsequent sector is successfully read, the controller decrements the UDC_SCNT register. If it is not 0, the controller increments the UDC_DSECT register and reads the next physical sector without regard to its sector number. This process continues until UDC_SCNT is reduced to 0, or an error occurs, or an index pulse is received from the drive. If the ID field of a sector about to be read has the BAD SECTOR bit set, the controller terminates the command with a TERMCOD value of 10 in the DKC_STAT port.

3.8.8.12 READ LOGICAL Command

The READ LOGICAL command reads one or more sectors from a track, beginning with a specified sector and continuing through logically consecutive sectors by incrementing the desired sector number until the sector count is satisfied or an unrecoverable error occurs. Figure 3-98 shows the READ LOGICAL command.

Figure 3-98: READ LOGICAL Command

7	6	5	4	3	2	1	0
0	1	0	1	1	1	BYPASS	XFER

Data Bit	Definition
BYPASS	Bypass bad sectors (bit 1). If this bit is 1, then the controller ignores any sectors marked with the BADSECT bit in the sector ID field. If the BYPASS bit is 0 and such a sector is encountered, the controller terminates the command with the TERMCOD field set to 10 and the BADSECT bit set to 1 in the DKC_STAT port.
XFER	Transfer data (bit 0). If this bit is 1, data is transferred from each sector into the disk data buffer. If it is 0, no data is transferred but all error checking is still performed.

Prior to executing this command, a drive must have been selected by a DRIVE SELECT command, the following registers must have been loaded, and the internal register pointer must be set to point to the UDC_DATA register.

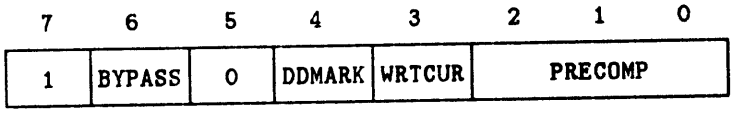
- UDC_MODE (mode appropriate to selected drive and media).
- UDC_DMAx (starting address in disk data buffer).
- UDC_DCYL (desired cylinder).
- UDC_DHEAD (desired head).
- UDC_DSECT (number of first sector).
- UDC_SCNT (number of sectors to be read).
- UDC_RTCNT (retry count).

The controller begins command execution by using the READ ID, VERIFY, and DATA TRANSFER sequences to find and read the first sector. After this and each subsequent sector is successfully read (possibly after retries), the controller decrements the UDC_SCNT register. If it is then not 0, the controller increments the UDC_DSECT register and uses the VERIFY and DATA TRANSFER sequences to find and read the next logical sector. This process continues until UDC_SCNT is reduced to 0 or an unrecoverable error occurs.

3.8.8.13 WRITE PHYSICAL Command

The WRITE PHYSICAL command writes one or more sectors on a track, beginning with a specified sector and continuing through physically consecutive sectors until the sector count is satisfied or the track index is encountered. Figure 3-99 shows the WRITE PHYSICAL command.

Figure 3-99: WRITE PHYSICAL Command



Data Bit	Definition
BYPASS	Bypass bad sectors (bit 6). If this bit is 1, then the controller ignores any sectors marked with the BADSECT bit in the sector ID field. If the BYPASS bit is 0 and such a sector is encountered, the controller terminates the command with the TERMCOD field set to 10 and the BADSECT bit set to 1 in the DKC_STAT port.
DDMARK	Deleted data mark (bit 4). If this bit is 1, the data is preceded by a deleted data mark (F8h). Otherwise the data is preceded by a normal data mark (FBh).
WRTCUR	Reduced write current (bit 3). Not used and must be 0.
PRECOMP	Write precompensation. Section 3.8.9 explains write precompensation.

Prior to executing this command, a drive must have been selected by a DRIVE SELECT command, the following registers must have been loaded, and the internal register pointer must be set to point to the UDC_DATA register.

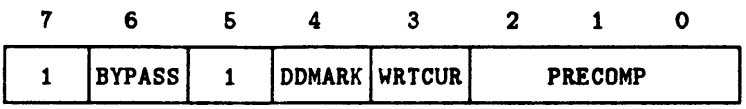
- UDC_MODE (mode appropriate to selected drive and media).
- UDC_DMAx (starting address in disk data buffer).
- UDC_DCYL (desired cylinder).
- UDC_DHEAD (desired head).
- UDC_DSECT (number of first sector).
- UDC_SCNT (number of sectors to be read).
- UDC_RTCNT (retry count. The RTRYCNT field must be set to 0 (1s complement form), since retries of write operations are not permitted).

The controller begins command execution by using the READ ID, VERIFY, and DATA TRANSFER sequences to find and write the first sector. After this and each subsequent sector is successfully written, the controller decrements the UDC_SCNT register. If it is not 0, the controller increments the UDC_DSECT register and writes the next physical sector without regard to its sector number. This process continues until UDC_SCNT is reduced to 0, an error occurs, or an index pulse is received from the drive.

3.8.8.14 WRITE LOGICAL Command

The WRITE LOGICAL command writes one or more sectors on a track, beginning with a specified sector and continuing through logically consecutive sectors by incrementing the desired sector number until the sector count is satisfied (see Figure 3-100).

Figure 3-100: WRITE LOGICAL Command



Data Bit	Definition
BYPASS	Bypass bad sectors (bit 6). If this bit is 1, then the controller ignores any sectors marked with the BADSECT bit in the sector ID field. If the BYPASS bit is 0 and such a sector is encountered, the controller terminates the command with the TERMCOD field set to 10 and the BADSECT bit set to 1 in the DKC_STAT port.
DDMARK	Deleted data mark (bit 4). If this bit is 1, the data is preceded by a deleted data mark (F8h). Otherwise the data is preceded by a normal data mark (FBh).
WRTCUR	Reduced write current (bit 3). Not used and must be 0.
PRECOMP	Write precompensation. Section 3.8.9 explains write precompensation.

Prior to executing this command, a drive must have been selected by a DRIVE SELECT command, the following registers must have been loaded, and the internal register pointer must be set to point to the UDC_DATA register.

- UDC_MODE (mode appropriate to selected drive and media).
- UDC_DMAx (starting address in disk data buffer).
- UDC_DCYL (desired cylinder).
- UDC_DHEAD (desired head).
- UDC_DSECT (number of first sector).
- UDC_SCNT (number of sectors to be read).

- UDC_RTCNT (retry count. The RTRYCNT field must be set to 0 (1s complement form), since retries of write operations are not permitted).

The controller begins command execution by using the READ ID, VERIFY, and DATA TRANSFER sequences to find and write the first sector. After this and each subsequent sector is successfully written, the controller decrements the UDC_SCNT register. If it is then not 0, the controller increments the UDC_DSECT register and uses the VERIFY and DATA TRANSFER sequences to find and write the next logical sector. This process continues until UDC_SCNT is reduced to 0 or an error occurs.

3.8.9 Write Precompensation

The FORMAT TRACK, WRITE PHYSICAL, and WRITE LOGICAL commands have a 3-bit field named PRECOMP in their command codes. The value of this field determines the amount of write precompensation applied to data which is written on a disk. The appropriate value depends upon the device type, media type, and what cylinder is being written. Table 3-30 lists the write precompensation parameters.

Table 3-30: Write Precompensation Parameters

Drive	Cylinders	Precomp	Time shift
RX33 diskette drive			
with RX33K media, 500 kHz	0..79	001	112 ns
with RX50K media, 250 kHz	0..79	100	212 ns
with 48tpi media, 250 kHz	0..39	100	212 ns
RD32 hard disk drive	0..819	000	none
RD53 hard disk drive	0..1023	000	none
RD54 hard disk drive	0..1225	000	none

3.8.10 Diskette Drive READY Condition

The drive status signal from an RX33 diskette drive serves as both a drive ready indicator and as a disk-changed indicator.

- The drive status signal is set to LOW when power is applied to the drive, and thereafter whenever the drive door latch is opened and the diskette is removed.

- The drive status signal is set to HIGH when a diskette is present, the door latch is closed, and a step pulse (in either direction) is sent to the drive.

When the drive has a diskette in it and is ready for operation, the drive status signal is HIGH. When the operator opens the door latch and removes the diskette (and when power is first applied), the status signal is set LOW. When the host program finds the status LOW, it should assume that any diskette which was previously in the drive has been removed. To find out whether another diskette has been inserted, the host program must issue 1 step pulse to the drive. If there is still no diskette in the drive, the status signal remains LOW, but if a new diskette has been inserted and the door latch has been closed, the drive is again ready for operation and the status signal becomes HIGH.

The drive status signal is visible to the host program in the READY bit of the UDC_DSTAT register. Table 3-31 shows the correspondence between the signal value and the READY bit value. This correspondence depends upon the INVRDY bit in the UDC_RTCNT register.

Table 3-31: Diskette Drive Status Signal

Drive Status	INVRDY Bit	READY Bit
LOW	0	1
HIGH	0	0
LOW	1	0
HIGH	1	1

The INVRDY bit is necessary since the controller chip does not issue any commands to the drive unless READY is 1. In order to issue normal seek, read and write commands and to issue a step command to attempt to change the drive status from LOW to HIGH, the host program must manipulate INVRDY to make READY a 1.

Before a host program performs an operation with an RX33 drive, it should test to see which of the following three states the drive is in.

1. Not Ready (no diskette present or just powered on)
2. Ready (diskette present and not changed since last test)
3. Changed (diskette present but possibly changed since last test).

The first step (after ensuring that the drive motor is running) is to set the INVRDY bit of UDC_RTCNT to 1, issue the DRIVE SELECT command, and examine the READY bit. If READY is 1, there is a diskette in the drive and it has not been changed since the last operation (state 2), so the program may continue with the operation.

If READY is 0, the door has been opened and the diskette has been removed since the last operation. The program should then clear INVRDY to 0, reselect the drive (to make the INVRDY change effective and set READY to 1), issue a STEP command to the drive (outward, unless it is at track 0 then it should be inward), and reexamine the READY bit. If the READY bit is now 0, there is a new diskette installed and the drive is ready (state 3). The host program can now change INVRDY back to 1, reselect the drive, and continue with its read or write operation. However, if READY is not 0 after the STEP command, there is either no diskette in the drive or the drive door is open (state 1).

NOTE: *For hard disks, INVRDY does not affect READY. However, INVRDY should be 0 for compatibility with early machines.*

3.8.11 Disk Programming

This section contains hints that programmers should be aware of when writing drivers for the disk controller.

3.8.11.1 Diskette Motor Control

The diskette drive motors are turned on and off by the MOTOR bit in the UDC_RTCNT register, and bit LOSPEED of that same register selects the rotation speed (300 or 360 rpm) as well as the data rate. Whenever the driver program starts the motor or changes its speed, the drive speed must be allowed to stabilize before the driver attempts to read from or write to the drive.

Production versions of the RX33 drive (p/n 30-24962-01, labeled "FD-55GFV-57-U") have an automatic lockout feature which enforces this timing restriction by suppressing read data from the drive until its motor speed is stable. If other diskette drives are used which do not have this lockout feature, then the driver software must provide a time delay after starting the motor or changing its speed. The time required must conform to the specifications of the drive being used. For example, the minimum times for prototype RX33 diskette drives without the lockout feature are listed in Table 3-32.

Table 3-32: RX33 Prototype Speed Change Timing Restrictions

Speed Changes	Timing Delay
Start to reach 300 rpm	400 milliseconds
Start to reach 360 rpm	500 milliseconds
Speed change (either way)	400 milliseconds

3.8.11.2 Implicit Seeks on Diskettes

After a seek operation moves a drive's heads, a settling time is required before the controller can receive stable data from the drive to verify the new head position and search for the desired sector. For hard disks, the drive determines this time by delaying its seek complete signal until its heads have settled. There is no such signal for diskette drives, so the controller chip attempts to read data immediately after issuing the last step pulse on diskette drives.

Production versions of the RX33 drive (p/n 30-24962-01, labeled "FD-55GFV-57-U") have an automatic lockout feature which enforces this timing restriction by suppressing read data from the drive until the head position has settled. If other diskette drives are used which do not have this lockout feature, then the driver software must insert a head settling delay time appropriate to the particular drive (for example, 18 milliseconds minimum for the prototype RX33 drives) after any head motion before attempting a write operation. Therefore, the driver must use a SEEK/READ ID command to move the heads to the desired track, wait for the delay time, and then issue the READ or WRITE command.

3.8.11.3 Diskette Write Completion Delay

At the conclusion of a WRITE PHYSICAL, WRITE LOGICAL, or FORMAT TRACK command (as signalled by the controller's DONE bit), the diskette drive requires some additional time to complete the tunnel erasure of the data just written. Therefore, a delay is required before doing any of the following:

- Moving the heads
- Deselecting the drive
- Changing the selected head number
- Stopping the motor

- Changing the motor speed.

It is important that driver programs observe this delay requirement since there is no hardware provision to enforce it. The minimum delay times for the RX33 drive are as follows:

High-speed (RX33 media) 0.59 milliseconds
Low-speed (RX50 media) 1.00 milliseconds.

3.8.11.4 Using the Disk and Tape Controllers

The 9224 disk controller chip, the 5380 tape controller chip, and the disk data buffer share a common local data bus which is used both by processor accesses to either chip or to the data buffer, and by DMA transfers between either chip and the data buffer. Therefore, it is not possible to use both the disk controller and the tape controller at the same time. Furthermore, whenever either controller has an outstanding DMA data transfer operation to or from the disk data buffer, the processor must not attempt to access the data buffer or any port in either controller chip until the chip signals that the current operation is done. Otherwise the processor access may collide with a DMA access cycle, which corrupts the data transfer for all parties.

One implication of this is that the interrupt system must be used by the controller chips to signal the completion of data transfer commands, since the processor cannot poll a controller chip during such a command.

3.8.11.5 Selecting the Diskette Drive

Whenever a DRIVE SELECT command selects the diskette drive and the drive has not been already selected, there may be a time delay before the controller can recover valid data from the drive. The disk data recovery circuit operates at two frequencies, one for hard disks and one for diskettes. It operates at the hard disk rate whenever either of the hard disk drives or no drive at all is selected, and at the diskette rate whenever the diskette drive is selected. When the transition from the hard disk rate to the diskette rate occurs, it takes up to 70 milliseconds for the data recovery circuit to stabilize at the lower speed.

The implications of this are that for efficient diskette operation, the diskette drive should remain selected between diskette sector accesses in order to keep the recovery circuit running at the diskette rate. Once selected, the diskette drive should not be deselected until a hard disk access is required or the diskette motor-on period expires. Otherwise, there may be missed diskette revolutions between consecutive operations on the diskette.

3.8.11.6 Drive Select Jumpers

The diskette drive (addressed by the controller as drive 10) is selected by drive select line 0 on pin 10 of its 34-pin connector. Therefore, the jumper plug on the drive should be inserted in position DS0 (this is the first of four positions). Diskette select lines are numbered from 0 through 3.

Both hard disk drives are selected by drive select line 3 on pin 30 of their 34-pin connectors. Therefore, the jumper plug on each drive should be inserted in position 3 (this is the third of four positions). The hard disk select lines are numbered from 1 through 4. The cabling between the system module and the drives maps controller address 00 to the drive in the system box and controller address 01 to the drive in the storage expansion box.

3.8.11.7 Spurious Data CRC Errors

The 9224 disk controller may indicate a spurious data CRC error when reading a diskette sector if it finds an apparent sync mark (bit pattern A1 hex with a missing clock bit) within approximately 16 bits following the end of the CRC bytes of the data sector. Such a spurious patterns can be created by diskette controllers on other systems which write only a one-byte pad following the CRC bytes (the 9224 disk controller writes a two-bytes pad), so this is primarily a system interchange problem.

The only solution is to not take diskette data CRC errors at face value. If a CRC error is signalled, the driver software should use the contents of the disk buffer, which will include the two CRC bytes following the data, to recompute the CRC. If the recomputed value matches the value in the buffer, the sector has been read correctly.

3.8.12 Diskette Drive Overview

The RX33 diskette drive uses 5.25-inch diskettes recorded in modified frequency modulation (MFM) mode and operates at two data rates; 250 KHz with standard RX50K media and 500 KHz with high-capacity RX33K media. The capacities of the diskette are listed in Table 3-33.

Table 3–33: Diskette Capacities

Item	Capacities/Speeds	
Number of tracks	80	
Number of heads	2	
Track density	96 tpi	
Track step rate	4 milliseconds per track	
Medium	RX50K	RX33K
MFM data bit rate	250 kHz	500 kHz
Rotation speed	300 rpm	360 rpm
512-byte sectors per track	10	15
Data capacity (1-sided)	400k bytes	
Data capacity (2-sided)		1200k bytes

The system supports 400k bytes on single sided RX50K media and 1200k bytes on double sided RX33K media. The system can format the tracks of an RX33K diskette, but it cannot format an RX50K diskette because the controller cannot omit the index address mark and its associated gaps. Another reason is because the drive speed tolerance is too great. Refer to the *RX33 Diskette Drive Technical Description Manual* (order number EK-RX33T-TM) for more information on the RX33 diskette drive.

3.8.13 Hard Disk Drives

The disk controller supports the hard disk drives using MFM recording at a data rate of 5 megabits per second. Each track is formatted to hold seventeen 512-byte sectors. A drive may have up to sixteen heads and up to 2048 cylinders. The drives that are supported are listed in Table 3-34.

Table 3–34: Hard Disk Capacities

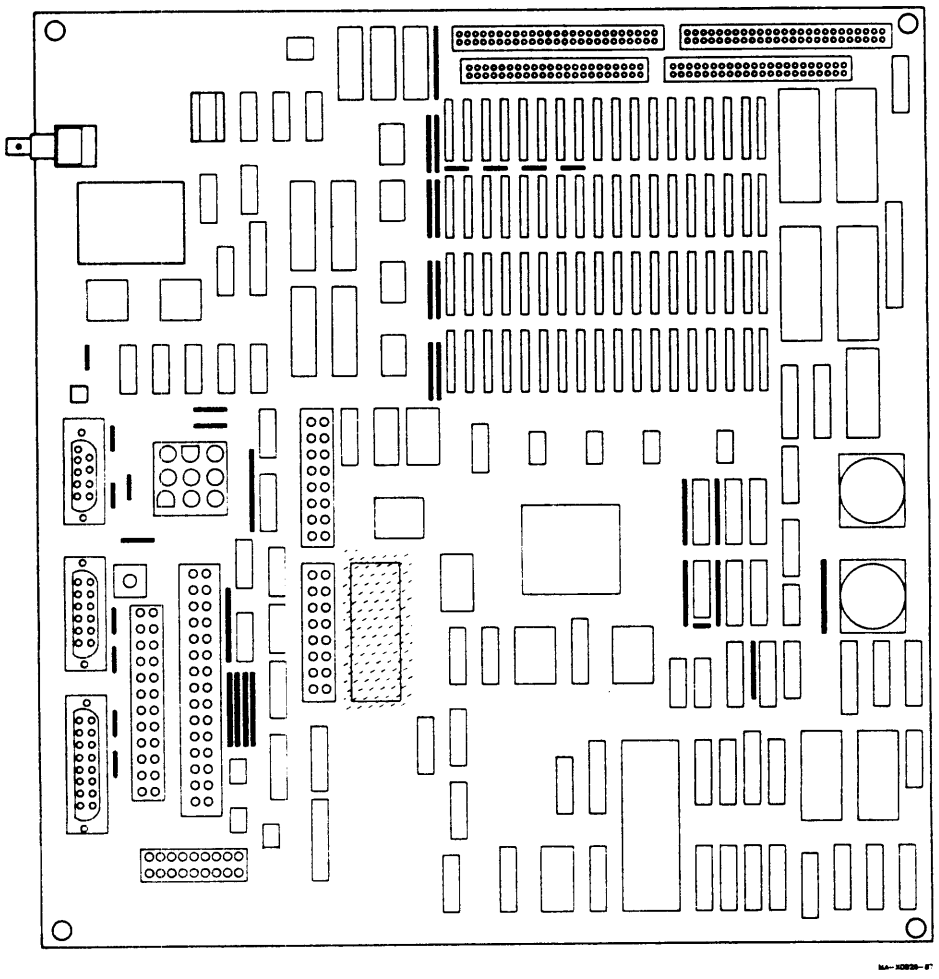
Item	Capacities/Speeds		
Model	RD32	RD53	RD54
Data bit rate	5 MHz	5 MHz	5 MHz
Rotation speed	3600 rpm	3600 rpm	3600 rpm
Capacity	41820K bytes	69632K bytes	156187K bytes
Cylinders	820	1024	1225
Heads	6	8	15
Average seek time	40 milliseconds	30 milliseconds	30 milliseconds

The RD32 drive is a half-height hard disk device. It can be installed in conjunction with one RX33 diskette drive in the system box. Two half-height hard disk drives cannot be installed because their combined motor starting surge during power-up exceeds the power supply capacity. The RD53 and RD54 drives are full-height devices. No other drive can be installed with either of these full-height drives in the system box or in the expansion box. Refer to the drive technical description manual for more information on the particular drive.

3.9 5380 Tape Controller

The 5380 tape controller (Figure 3-101) provides an ANSI small computer system interface (SCSI) between the TZK50 tape controller in the tape expansion box and the data buffer on the system module.

Figure 3-101: 5380 Tape Controller



The following sections provide the theory of operation of the tape controller, an overview of the SCSI tape bus operation, a breakdown of the registers that control the tape controller, and an explanation of the conditions that generate an interrupt.

- 5380 Tape Controller Overview (Section 3.9.1)
- SCSI Overview (Section 3.9.2)
- 5380 Tape Controller Chip Register (Section 3.9.3)
- DMA Register Operation (Section 3.9.4)
- Tape Controller Interrupt (Section 3.9.5)

3.9.1 5380 Tape Controller Overview

The tape controller is an NCR 5380 SCSI controller chip. It is connected directly to the SCSI tape bus (port A on the expansion adapter), and it is also connected to the disk data buffer via the disk buffer data bus. The 5380 is controlled by the DC524 standard cell. Figure 3-102 shows a circuit diagram of the 5380 tape controller chip and Table 3-35 lists a description of its signals.

Figure 3-102: 5380 Tape Controller Chip Pinout

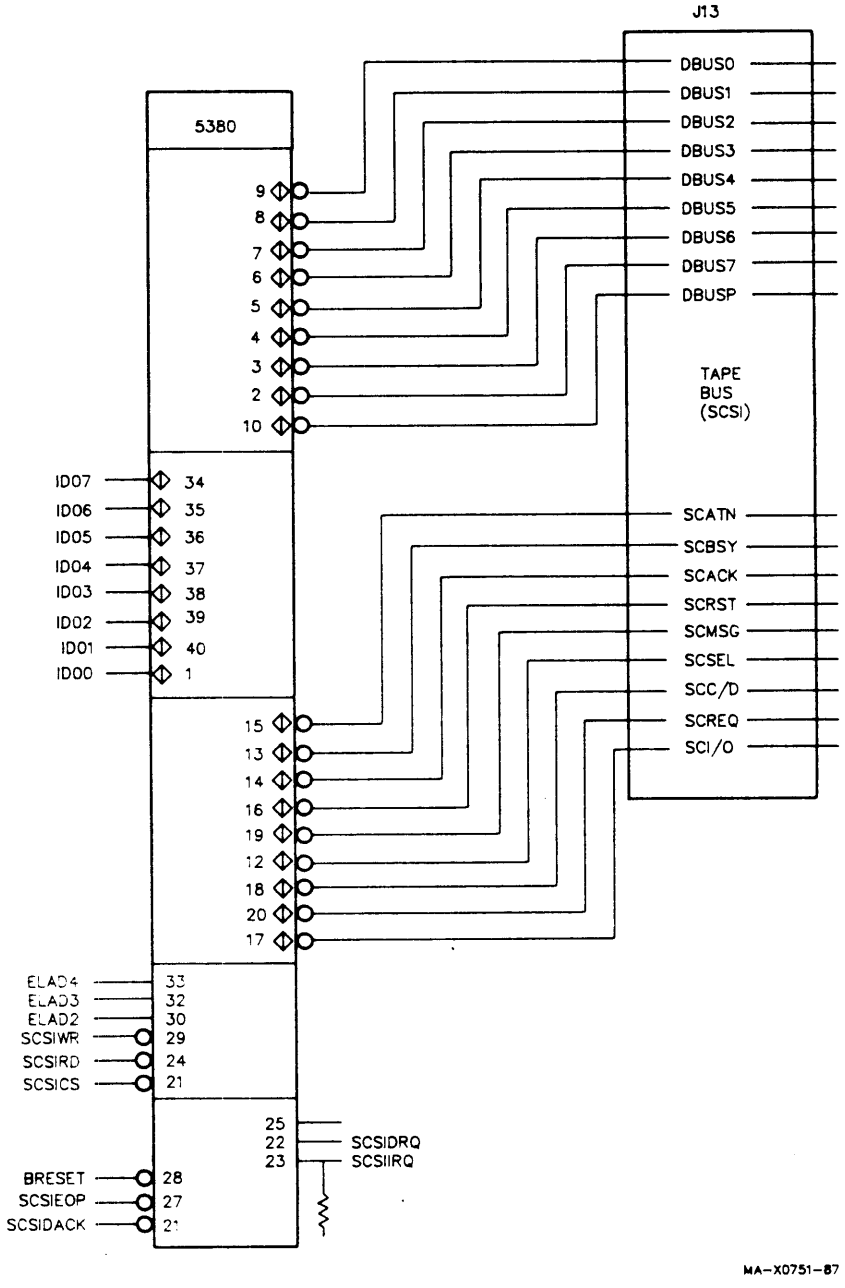


Table 3–35: 5380 Tape Controller Chip Pinout

Pin	Signal	Description
2:9	DBUS7:0	These signals are the SCSI data bus. The SCSI data bus transfers data to and from the 5380 controller and the TZK50 tape controller in the tape expansion box.
10	DBUSP	This signal is the SCSI data bus parity bit.
34:40 1	ID07:1 ID00	These signals are the internal data bus, which transfers data to and from the disk data buffer or to and from the CPU chip.
15	SCATN	This signal is the attention bit on the SCSI tape bus.
14	SCACK	This signal is the acknowledge bit on the SCSI tape bus.
13	SCBSY	This signal is the busy bit on the SCSI tape bus.
12	SCSEL	This signal is the select bit on the SCSI tape bus.
16	SCRST	This signal is the reset bit on the SCSI tape bus.
17	SCI/O	This signal is the input/output bit on the SCSI tape bus.
18	SCC/D	This signal is the command/data bit on the SCSI tape bus.
19	SCMSG	This signal is the message bit on the SCSI tape bus.
20	SCREQ	This signal is the request bit on the SCSI tape bus.
33:31	ELAD4:2	These signals are address lines used by the CPU and standard cell when initializing the 5380 for a DMA transfer.
29	SCSIWR	This signal is the write control strobe from the standard cell.
24	SCCSIRD	This signal is the read control strobe from the standard cell.
21	SCSICS	This signal is the chip select control line from the standard cell.
25	READY	This signal is not used.
22	SCSIDRQ	This signal is the DMA request line to the standard cell.

Table 3–35 (Cont.): 5380 Tape Controller Chip Pinout

Pin	Signal	Description
23	SCSIIRQ	This signal is the interrupt request line to the standard cell.
28	BRESET	This signal is the reset line from the CPU chip. It is used during power-up to initialize the 5380.
27	SCSIEOP	This signal is the end of process indicator.
26	SCSIDACK	This signal is the DMA acknowledge line from the standard cell.

The disk data buffer is used by both the 9224 disk controller and the 5380 controller during data transfer. The 5380 uses the lower byte (ID07:ID00) of the disk buffer data bus to transfer data to and from the disk data buffer and the SCSI tape bus. When the 5380 needs to transfer data, the CPU isolates the disk buffer data bus from the other buses by holding all bus transceivers in the high impedance state until the transfer is complete. Only one device can access the data buffer at the same time. Device driver software must ensure that only one device is allowed to access the data buffer at the same time.

The SCSI tape bus contains eight data bus lines, including one parity line, and nine control lines.

A host program can examine and manipulate all the SCSI signals using the 5380 chip. Associated with the chip is DMA logic, which can transfer data between the SCSI tape bus and the disk data buffer. The normal method of operation is for the host program to do programmed data transfers for the command, status, and message phases, which handle only a few bytes at a time, and to set up DMA transfers for the data phases. The 5380 chip can be used by both initiator and target devices. In this manual, only its use as an initiator is described.

3.9.2 SCSI Overview

The SCSI electrical and logical interface and operation is described in detail in the ANSI draft standard issued by ANSI task group X3T9.2, and the particular subset of that standard used by the tape controller is described in the TZK50 specification. The programmer must use both of those documents in conjunction with this specification as his guide. This section reviews a few important features of the ANSI document to set the context for the following discussion of the VS410 implementation of the SCSI interface.

The SCSI interface is a bi-directional 8-bit-wide bus to which up to eight devices can be attached. The system module is one of those devices, so up to seven additional devices can be attached. Devices may play one of two roles: initiator or target. An initiator originates an operation by sending a command to a specific target. A target performs an operation which was requested by an initiator. In this specification it is assumed that the system module is always an initiator and that all other SCSI devices attached to it are targets. (There is, however, no hardware feature of the system module SCSI interface which prevents its sharing the bus with a second initiator or assuming the role of a target.)

Each device attached to the SCSI tape bus is identified by a unique device ID number in the range 0 through 7. During the arbitration, selection, and reselection bus phases in which an initiator and a target establish a connection, the device IDs of the initiator and target are both placed on the data bus by asserting the data bits corresponding to the device ID numbers. By convention, the ID number of the system module is 0. (The ID number of the system module is controlled by the program which drives the SCSI interface. It is not fixed in system module hardware).

The electrical interface consists of 18 signal lines on a 50-pin connector. Some of these lines are driven only by initiators, others only by targets, and others by both initiators and targets. These 18 SCSI tape bus signal lines are summarized in Table 3-36. The signal names are the same as those described in the ANSI specification. Table 3-37 lists information transfer phases associated with the C/D, I/O, and MSG tape bus signals.

In all the registers of the 5380 controller chip, the true or asserted value of a signal appears as a 1, and the false or negated value of a signal appears as a 0. The bus electrical signals are all low true and are driven by open-collector drivers.

Table 3-36: SCSI Tape Bus Signal Definitions

Signals	Definitions
DB7:0 and DBP	These signals are an 8-bit parallel data bus with an associated odd parity bit. The use of the parity bit is optional but strongly encouraged. These lines may be driven by either an initiator or a terminator, depending on the direction of data transfer.
RST	This signal flags all devices on the SCSI tape bus to reset to their initial power-on states. The system firmware asserts this signal at least once during power-on self-test. Thereafter, it should be asserted only as a last resort during error recovery since it affects all devices on the bus. An RST signal generated by some other device on the bus causes an internal reset of the 5380 chip and sets the interrupt request bit (INTREQ in register SCS_STATUS).
BSY and SEL	These signals are used by initators and targets during the arbitration, selection, and reselection bus phases to establish or resume a logical connection between an initiator and a target. Once the connection is established, the target asserts BSY and the SEL signal is dropped.
C/D, I/O and MSG	These signals collectively indicate one of six possible information transfer phases (see Table 3-37). The signals in Table 3-37 are always driven by the target device.
ATN	This signal is used by an initiator to signal a target that it has a message ready. The target can receive the message by entering the message out phase. ATN is always driven by an initiator.
REQ and ACK	These signals are used to synchronize information transfers over the data bus during any of the six information transfer phases. REQ is always driven by the sender of the information after it has placed data on the DB7..0 and DBP lines. ACK is driven by the receiver of the information after it has captured it from the data lines. The system module supports only asynchronous data transfer in which a sender may assert REQ only once before receiving ACK from the receiver. The synchronous option is not supported.

Table 3–37: SCSI Tape Bus Information Transfer Phases

MSG	C/D	I/O	Phase Name	Transfer Direction
0	0	0	Data out	To target
0	0	1	Data in	To initiator
0	1	0	Command	To target
0	1	1	Status	To initiator
1	0	0	(reserved)	
1	0	1	(reserved)	
1	1	0	Message out	To target
1	1	1	Message in	To initiator

3.9.3 5380 Tape Controller Chip Registers

The controller chip appears to the system as a group of thirteen 8-bit registers which are addressed on longword boundaries. Nine of these registers contain data bits which can be read and/or written by a host program. The remaining four have no data bits but are action registers. This means that when the host program reads or writes one of them, the controller chip is signalled to take some action, but the data bits are ignored. Table 3–38 lists the thirteen registers in the 5380 chip.

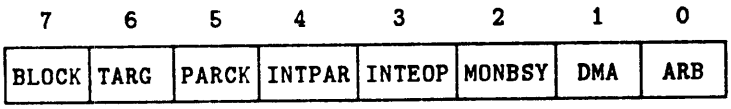
Table 3-38: 5380 Controller Chip Register Addresses

Address	Name	Access	Description
200C.0088	SCS_MODE	r/w	Mode register
200C.0084	SCS_INI_CMD	r/w	Initiator command register
200C.008C	SCS_TAR_CMD	r/w	Target command register
200C.0094	SCS_STATUS	r	Bus and status register
200C.0090	SCS_CUR_STAT	r	Current bus status register
200C.0090	SCS_SEL_ENA	w	Select enable register
200C.0080	SCS_OUT_DATA	w	Output data register
200C.0080	SCS_CUR_DATA	r	Current data register
200C.0098	SCS_IN_DATA	r	Input data register
200C.0094	SCS_DMA_SEND	w	Start DMA send action
200C.009C	SCS_DMA_IRCV	w	Start DMA initiator receive action
200C.0098	SCS_DMA_TRCV	w	Start DMA target receive action
200C.009C	SCS_RESET	r	Reset interrupt/error action

3.9.3.1 Mode Register (SCS_MODE)

The mode register is an 8-bit read/write register at physical address 200C.0088 that controls the operation of the chip. This register determines whether the system operates as an initiator or target, whether DMA transfers are being used, whether parity is checked for SCSI tape bus data, and whether interrupts are signalled for various conditions. Figure 3-103 shows the mode register.

Figure 3-103: Mode Register (SCS_MODE)



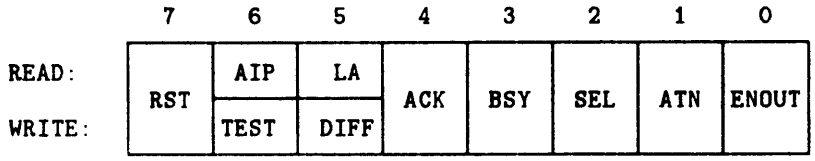
Data Bit	Definition
BLOCK	DMA block mode (bit 7). This bit controls the characteristics of the DRQ-DACK handshake between the 5380 chip and the DMA controller during DMA data transfers. For the system module, this bit must always be 0.
TARG	Target role (bit 6). This bit determines whether the system performs the role of an initiator (TARG is 0) or target (TARG is 1) on the SCSI tape bus. The system module normally acts as an initiator, so this bit should normally be 0.
PARCK	Parity check enable (bit 5). This bit determines whether SCSI tape bus data parity errors are ignored (PARCK is 0) or enabled (PARCK is 1). If this bit is 1 and a parity error is detected, the PARERR bit in the SCS_STATUS register is also set to 1.
INTPAR	Interrupt on parity error (bit 4). This bit determines whether parity errors detected on the SCSI tape bus signal an interrupt. BOTH INTPAR and PARCK need to be 1 when an error is detected to signal an interrupt.
INTEOP	Interrupt on end of DMA (bit 3). This bit determines whether an interrupt is generated at the end of a DMA transfer. If INTEOP is 1, an interrupt is signalled when the DMA count register SCD_CNT reaches 0.
MONBSY	Monitor BSY (bit 2). While this bit is 1, the chip signals an interrupt upon a loss of the bus BSY signal. When such an interrupt is generated, bits 5..0 of the SCS_INI_CMD register (bits DIFF, ACK, BSY, SEL, ATN, and ENOUT) are cleared to 0. This removes all signals generated by the system from the SCSI tape bus.
DMA	Enable DMA transfer (bit 1). This bit, when set to 1, enables DMA transfers between the controller chip and the disk buffer. This bit must be set to 0 for programmed data transfers. Setting this bit to 0 also clears the DMAEND bit in the SCS_STATUS register. This bit must be set as part of the DMA initialization process which also includes initializing the DMA controller registers SCD_ADR, SCD_CNT, and SCD_DIR, and appropriately setting the ENOUT bit of the SCS_INI_CMD register. After this initialization, the host program must write the appropriate action register (SCS_DMA_SEND or SCS_DMA_IRCV) to begin actual data transfers.

Data Bit	Definition
	The DMA bit is not cleared when the DMA count register SCD_CNT reaches 0. It must be cleared by the host program. Once this bit is cleared, no further DMA data transfers occur. **NOTE:** *The host cannot reliably stop an ongoing DMA transfer by clearing this bit because the chip's data path may be in use for a DMA transfer. If so, the host may not able to access the chip's registers.*
ARB	Start arbitration (bit 0). The host program sets this bit to 1 to start the bus arbitration process. Prior to setting this bit the program should load the system's device ID (conventionally bit 0) in the output data register SCS_OUT_DATA. The chip waits for a bus-free condition before entering the arbitration phase. The results of the arbitration may be determined by reading bits AIP and LA of the initiator command register SCS_INI_CMD.

3.9.3.2 Initiator Command Register (SCS_INI_CMD)

The initiator command register is an 8-bit read/write register at physical address 200C.0084. It is used when the system is acting as an initiator (its normal role) to assert certain SCSI tape bus control signals, to monitor those signals, and to monitor the progress of bus arbitration. Bits 5 and 6 of this register have different definitions according to whether the register is read from or written to. Therefore, programs cannot use read-modify-write instructions such as BISB and BICB to access this register. Figure 3-104 shows the initiator command register.

Figure 3-104: Initiator Command Register



Data Bit	Definition
RST	Assert RST (bit 7, read/write). When this bit is changed from 0 to 1 (by writing to the SCS_INI_CMD register), the RST signal is asserted, the chip interrupt request signal is asserted, and all the chip's internal logic and control registers are cleared (except for this bit and the interrupt request latch). While this bit is 1, the RST signal is asserted on the SCSI tape bus. Writing a 0 to this bit negates the RST signal. Reading this bit reflects only its value in the SCS_INI_CMD register and not necessarily the actual bus signal state.
AIP	Arbitration in progress (bit 6, read-only). This bit, when set to 1, indicates the bus arbitration is in progress. In order for this bit to be set, the ARB bit in SCS_MODE must also be set. A 1 in the AIP bit indicates that a bus free condition has been detected and that the the chip has asserted BSY and the contents of the SCS_OUT_DATA register onto the SCSI tape bus. AIP remains set until the ARB bit in SCS_MODE is cleared.
TEST	Test mode (bit 6, write-only). Setting this bit to 1 disables all the chip output drivers, effectively removing the system module from the SCSI tape bus. Note that setting this bit may generate spurious DMA requests or interrupts to the CPU chip; therefore, the use of this bit is not recommended. This bit must be 0 for normal operation.
LA	Lost arbitration (bit 5, read-only). This bit, when set to 1, indicates that the chip detected a bus-free condition, arbitrated for use of the bus by asserting BSY and the system's ID (in SCS_OUT_DATA) on the bus, but lost the arbitration because SEL was asserted by some other device on the bus. The LA bit can only be asserted while the ARB bit of SCS_MODE is set.
DIFF	Differential enable (bit 5, write-only). Must always be 0 in this system.
ACK	Assert ACK (bit 4, read/write). While this bit is 1, the ACK signal is asserted on the SCSI tape bus. This bit is effective only while the TARG bit of SCS_MODE is 0 (that is, when the system is acting as an initiator). Writing a 0 to the ACK bit negates the ACK signal. Reading this bit reflects only its value in the SCS_INI_CMD register and not necessarily the actual bus signal state.
BSY	Assert BSY (bit 3, read/write). While this bit is 1, the BSY signal is asserted on the SCSI tape bus. Asserting BSY indicates a successful selection or reselection. Writing a 0 into the BSY bit negates the BSY signal, which indicates a bus disconnect condition. Reading this bit reflects only its value in the SCS_INI_CMD register and not necessarily the actual bus signal state.

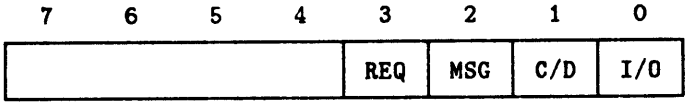
Data Bit	Definition
SEL	Assert SEL (bit 2, read/write). While this bit is 1, the SEL signal is asserted on the SCSI tape bus. SEL is normally asserted after arbitration has been successfully completed. Writing a 0 into the SEL bit negates the SEL signal. Reading this bit reflects only its value in the SCS_INI_CMD register and not necessarily the actual bus signal state.
ATN	Assert ATN (bit 1, read/write). While this bit is 1, the ATN signal is asserted on the SCSI tape bus. This bit is effective only while the TARG bit of SCS_MODE is 0 (that is, when the system is acting as an initiator). Writing a 0 to the ATN bit negates the ATN signal. Reading this bit reflects only its value in the SCS_INI_CMD register and not necessarily the actual bus signal state.
ENOUT	Enable output (bit 0, read/write). This bit, when set to 1, allows the contents of the SCS_OUT_DATA register to be sent out on the SCSI tape bus. When operated as an initiator (i.e. the TARG bit of SCS_MODE is 0), the outputs are only enabled while the bus I/O signal is false and the three bus phase signals C/D, I/O and MSG match the contents of the corresponding bits in the SCS_TAR_CMD register. The ENOUT bit must be set to 1 during DMA operations which send data out to the SCSI tape bus.

3.9.3.3 Target Command Register (SCS_TAR_CMD)

The target command register is an 8-bit read/write register at physical address 200C.008C. When the system is acting as an initiator (its normal role), this register is used during DMA data transfers (i.e. when bit DMA of the SCS_MODE register is 1) to monitor the bus phase. When a target asserts REQ to request a data transfer, if the state of the bus MSG, C/D and I/O signals does not match the values of those bits in this register, a phase mismatch interrupt is generated. This enables the host program to be notified when a DMA data transfer is ended by the target (this may occur prior to the DMA counter's reaching 0, since the target controls the length of data transfers). In initiator mode, the REQ bit in this register is ignored.

When the system is used as a target device (the TARG bit in SCS_MODE is 1), this register allows a program to assert the REQ, MSG, C/D and I/O signals on the SCSI tape bus. Figure 3-105 shows the target command register.

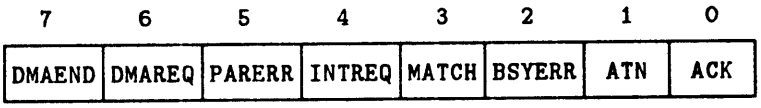
Figure 3-105: Target Command Register



3.9.3.4 Bus and Status Register (SCS_STATUS)

The bus and status register is an 8-bit read-only register at physical address 200C.0094. It contains six chip status flags and monitors two of the SCSI tape bus control signals, ACK and ATN. The other seven bus control signals are visible in the current bus status register (SCS_CUR_STAT). Figure 3-106 shows the SCSI tape bus and status register.

Figure 3-106: SCSI Tape Bus and Status Register

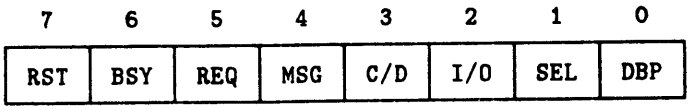


Data Bit	Definition
DMAEND	DMA end (bit 7). This bit is set when the DMA count register SCD_CNT becomes 0 during a data transfer. After this bit is set, the chip performs no additional DMA cycles. The DMAEND bit is cleared when the DMA bit in the SCS_MODE register is cleared.
DMAREQ	DMA request (bit 6). This pin reflects the status of the internal DMA request signal from the 5380 chip to the DMA controller. This bit becomes 1 when the chip requests the transfer of a byte to or from the disk buffer, and returns to 0 when the DMA controller has performed the transfer.
PARERR	Parity error (bit 5). This bit is set upon receipt of a byte with incorrect parity from the SCSI tape bus during a data transfer to the system or during device selection. PARERR is set only if the PARCK bit of the SCS_MODE register is set to 1 to enable parity checking. PARERR is not set while PARCK is 0. The PARERR bit is cleared when the reset interrupt/error register SCS_RESET is read.
INTREQ	Interrupt request (bit 4). This bit is set when any of the interrupt conditions described in Section 3.9.5 occurs. It is cleared when the reset interrupt/error register SCS_RESET is read.
MATCH	Phase match (bit 3). This bit is 1 whenever the three SCSI tape bus phase signals MSG, C/D, and I/O match the values in the corresponding three bits of the target command register SCS_TAR_CMD. The MATCH bit is continuously updated and is only significant when the system is operating as an initiator (its normal mode). MATCH must be 1 for data transfers to occur on the SCSI tape bus.
BSYERR	Busy error (bit 2). This bit is set whenever the MONBSY bit of the mode register SCS_MODE is 1 and the SCSI tape bus BSY signal is false. This feature is used to monitor the bus for an unexpected loss of the logical connection between the system (as initiator) and a target device. When BSYERR is set, the DMA bit in the mode register SCS_MODE is cleared to stop any DMA data transfers, and the DIFF, ACK, BSY, SEL, ATN and ENOUT bits of the SCS_INI_CMD register are cleared to remove all signals generated by the system from the SCSI tape bus.
ATN	ATN signal (bit 1). This bit reflects the current state of the ATN signal on the SCSI tape bus.
ACK	ACK signal (bit 0). This bit reflects the current state of the ACK signal on the SCSI tape bus.

3.9.3.5 Current Bus Status Register (SCS_CUR_STAT)

The current bus status register is an 8-bit read-only register at physical address 200C.0090. It is used to monitor seven of the SCSI tape bus control signals plus the data bus parity bit. The other two bus control signals, ACK and ATN, are in the bus and status register (SCS_STATUS). The host program uses the current bus status register to determine the current bus phase and to poll REQ during programmed data transfers from the system to a target device. This register is also used to help determine why a particular interrupt occurred. Figure 3-107 shows the current bus status register.

Figure 3-107: Current Bus Status Register

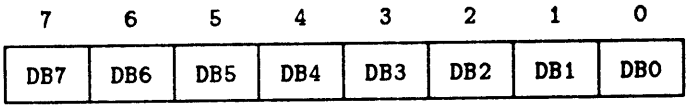


3.9.3.6 Select Enable Register (SCS_SEL_ENA)

The select enable register is an 8-bit write-only register at physical address 200C.0090. It contains the device ID of the system module. The system module should recognize this ID as its own during a selection or reselection attempt. For a VS410 system whose device ID is normally 0 this register should contain a 1 in bit 0 and 0's elsewhere.

The simultaneous occurrence of the correct ID bit on the data bus, BSY false, and SEL true (during a selection or reselection phase) generates an interrupt signal. Such interrupts can be disabled by writing all 0's into this register. If parity checking is enabled (the PARCK bit in the mode register SCS_MODE is set), the parity of the data on the data bus is checked during selection or reselection. Figure 3-108 shows the select enable register.

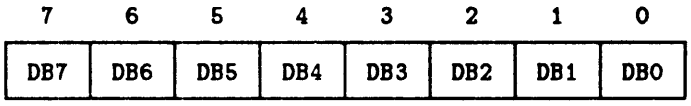
Figure 3-108: Select Enable Register (SCS_SEL_ENA)



3.9.3.7 Output Data Register (SCS_OUT_DATA)

The output data register is an 8-bit write-only register at physical address 200C.0080. It is used to send outgoing data to the SCSI tape bus. It is used during programmed I/O to write outgoing data bytes and to assert the proper ID bits on the SCSI tape bus during arbitration and selection phases. This register is also implicitly used by the hardware during DMA transfers to the SCSI tape bus. Figure 3-109 show the output data register.

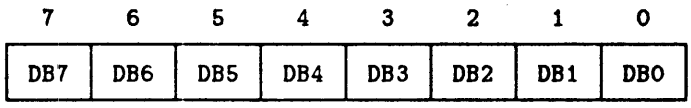
Figure 3-109: Output Data Register



3.9.3.8 Current Data Register (SCS_CUR_DATA)

The current data register is an 8-bit read-only register at physical address 200C.0080. Its contents reflect the data currently on the data lines of the SCSI tape bus. It is used during programmed (rather than DMA) I/O to read incoming data bytes and during arbitration to check for higher priority arbitrating devices. Figure 3-110 show the current data register.

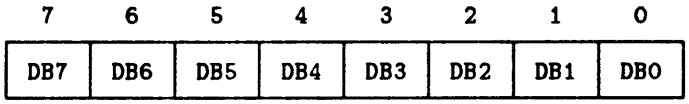
Figure 3-110: Current Data Register (SCS_CUR_DATA)



3.9.3.9 Input Data Register (SCS_IN_DATA)

The input data register is an 8-bit read-only register at physical address 200C.0098. It is used to read latched data from the SCSI tape bus during DMA operation. During programmed I/O operation, no data is latched in this register. The SCS_CUR_DATA register should be used instead for programmed I/O. The input data register is implicitly used by the hardware during DMA transfers from the SCSI tape bus. When the system is acting as an initiator (its normal role), data is latched when the bus REQ signal is asserted by the target device. When the system is acting as a target, data is latched when the bus ACK signal is asserted. Figure 3-111 shows the input data register.

Figure 3-111: Input Data Register (SCS_IN_DATA)



3.9.3.10 Start DMA Send Action (SCS_DMA_SEND)

The start DMA send action register is an 8-bit write-only register at physical address 200C.0094. The act of writing to this register begins DMA transfers from the system disk buffer to a target device. The data written to this register is ignored. Prior to writing to this register, the DMA controller registers SCD_ADR and SCD_CNT must be loaded, the DIR bit of the SCD_DIR register must be set to 0, the ENOUT bit of the INI_CMD register must be set to 1, and the DMA bit of the SCS_MODE register must be set to 1.

3.9.3.11 Start DMA Initiator Receive Action (SCS_DMA_IRCV)

The start DMA initiator receive action register is an 8-bit write-only register at physical address 200C.009C. The act of writing to this register begins DMA transfers from a target on the bus to the system disk buffer, when the system is acting as an initiator device (its normal role). The data written to this register is ignored. Prior to writing to this register, the DMA controller registers SCD_ADR and SCD_CNT must be loaded, the DIR bit of the SCD_DIR register must be set to 1, the ENOUT bit of the INI_CMD register must be set to 0, and the DMA bit of the SCS_MODE register must be set to 1.)

3.9.3.12 Start DMA Target Receive Action (SCS_DMA_TRCV)

The start DMA target receive action register is an 8-bit write-only register at physical address 200C.0098. The act of writing to this register begins DMA transfers from an initiator on the bus to the system disk buffer, when the system is acting as a target device (not its normal role). The data written to this register is ignored. Prior to writing to this register, the DMA controller registers SCD_ADR and SCD_CNT must be loaded, the DIR bit of the SCD_DIR register must be set to 1, the ENOUT bit of the INI_CMD register must be set to 0, and the DMA bit of the SCS_MODE register must be set to 1.

3.9.3.13 Reset Interrupt/Error Action (SCS_RESET)

The reset interrupt/error action register is an 8-bit read-only register at physical address 200C.009C. The act of reading this register clears bits PARERR (parity error), INTREQ (interrupt request), and BSYERR (busy error) in the bus and status register SCS_STATUS. No useful data is returned from reading this register.

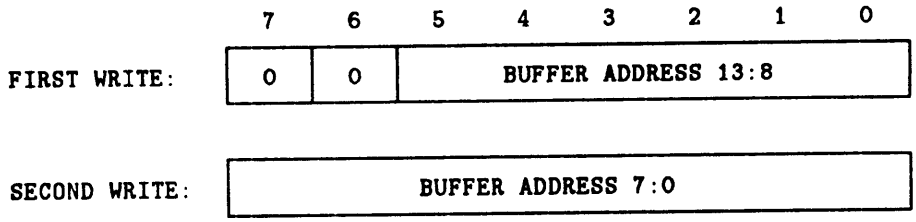
3.9.4 DMA Register Operation

This section describes registers associated with the DMA transfer operation.

3.9.4.1 DMA Address Register (SCD_ADR)

The DMA address register is an 8-bit write-only register at physical address 200C.00A0. It is used to set the starting address in the disk buffer for the next DMA transfer. Figure 3-112 shows the DMA address register.

Figure 3-112: DMA Address Register (SCD_ADR)



Since the disk buffer contains 16K bytes, the required 14-bit starting address must be loaded by two consecutive writes to SCD_ADR. The first write sets bits 13:8 of the starting address from bits 5:0 of the data byte. The second write sets address bits 7:0 from bits 7:0 of the second data byte.

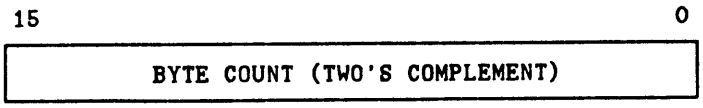
This register must not be accessed while a DMA operation is either pending or in progress for either the tape controller or the disk controller, since both controllers use the same address register.

NOTE: *Two consecutive writes to SCD_ADR always load the address correctly, even if a previous single write to SCD_ADR only partially sets the address. Each write to SCD_ADR moves the contents of buffer address bits 7:0 into bits 15:8 and then loads bits 7:0 from the data presented by the write operation.*

3.9.4.2 DMA Count Register (SCD_CNT)

The DMA count register is a 16-bit read/write register at physical address 200C.00C0. It counts the number of bytes transferred during a DMA operation and signals the tape controller chip when the specified number of bytes have been transferred. This register should be loaded with the 16-bit 2's complement of the maximum number of bytes to be transferred by the next DMA transfer between the tape controller chip and the disk data buffer. This counter is not used for and is not affected by DMA operations between the disk controller and the disk buffer. Figure 3-113 shows the DMA count register.

Figure 3-113: DMA Count Register (SCD_CNT)



As each byte is transferred, SCD_CNT is incremented by 1. When SCD_CNT changes from -1 to 0, a counter overflow bit is set and the tape controller chip is signalled through its EOP pin to terminate DMA operation at the completion of the current byte transfer (that is, the transfer during which the counter becomes 0). This sets the DMAEND bit in the SCS_STATUS register.

While the counter overflow bit is set, the DMA controller does not perform data transfers. The counter overflow bit is cleared whenever the count register is loaded by writing to SCD_CNT. If a transfer request is pending (the DMAREQ bit of the SCS_STATUS register is true) when the counter is loaded, a transfer occurs at once. Therefore, when restarting a DMA transfer, the host program must first load SCD_ADR and SCD_DIR and then load SCD_CNT with a single word write.

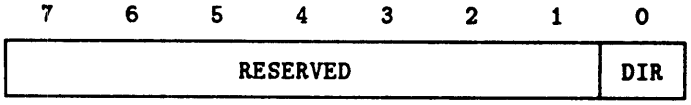
After a DMA operation ends (either because SCD_CNT reached 0 or because the tape contoller chip sensed a bus phase change), the host program may read SCD_CNT to get the number (in 2's complement form) of bytes not transferred. Adding this to the true form of the count originally loaded into SCD_CNT gives the number of bytes actually transferred. This register must not be read or written while an DMA operation is either pending or in progress. At power-on, SCD_CNT and its overflow bit are cleared to 0.

NOTE: *There is an interaction between the SCD_CNT register and the HLTCOD register. The contents of HLTCOD must be 0 whenever a program attempts to read the contents of SCD_CNT; otherwise the value received may be in error. The contents of HLTCOD do not affect program writes to SCD_CNT, and do not affect actual DMA operation.*

3.9.4.3 DMA Direction Register (SCD_DIR)

The DMA direction register is an 8-bit write-only register at physical address 200C.00C4. It controls the direction in which data is transferred during DMA cycles requested by the tape controller chip. Figure 3-114 shows the DMA direction register.

Figure 3-114: DMA Direction Register



Data Bit	Definition
7:1	Reserved. Must always be written as 0's.
DIR	Transfer direction (bit 0). When this bit is 1, DMA cycles transfer data from the SCSI tape bus into the disk buffer (a READ operation). When this bit is 0, DMA cycles transfer data from the disk buffer to the SCSI tape bus (a WRITE operation). Upon power-on, DIR is cleared to 0.

3.9.5 Tape Controller Interrupts

The 5380 chip has one interrupt request signal which is sent to the CPU through the system interrupt controller. The state of this signal is visible in the INTREQ bit of the SCS_STATUS register. An interrupt request can be signalled by any of the following six events.

- The controller is selected or reselected by another device on the SCSI tape bus.
- The DMA count register reaches 0.
- A parity error is detected during data transfer.
- A bus phase mismatch occurs.
- An SCSI tape bus disconnect occurs.
- The RST signal is asserted on the SCSI tape bus.

When the host program responds to the interrupt, it must use the contents of the SCS_STATUS and SCS_CUR_STAT registers to determine what condition(s) caused the interrupt. Once it has serviced the interrupt, the host program must read the SCS_RESET register to reset the INTREQ bit. Each of the above interrupts cause an except receipt of the RST signal and can be individually masked by appropriate settings of the 5380 chip registers.

In order for the 5380 interrupt signal to cause a CPU interrupt, the SC bit of the interrupt mask register INT_MSK must be set. Section 3.5.9.4 lists the value of the tape controller's interrupt vector.

3.9.5.1 Selection or Reselection

An interrupt can be signalled when another device on the SCSI tape bus attempts to select or reselect the system module. In the system's normal role of an initiator, the system module may be reselected by a device to which it has previously issued a command. Selection is appropriate only if the system is acting as a target. Such an interrupt occurs when the following conditions are met.

- The SEL signal is true.
- The BSY signal is false for at least a bus settle delay (400 ns).
- The logical AND of each data bus bit, DB7:0, with the corresponding bit in the select enable register, SCS_SEL_ENA, are 1.

The interrupt service routine can identify this type of interrupt by noting that BSY is 0 and SEL is 1 in the SCS_CUR_STAT register. If the I/O bit of SCS_CUR_STAT is 0, this is a select attempt. Otherwise it is a reselect. The SCS_SEL_ENA register should contain a 1 only in the bit corresponding to the system's SCSI device ID (normally bit 0) and 0's in the other seven bits. Only two bits should be asserted on the SCSI data bus during selection or reselection; they are the ID of the initiator and the ID of the target. The host program should check this by examining the data bus through the SCS_CUR_DATA register. In addition, if bus parity is enabled (bit PARCK of SCS_MODE is true), then the parity error bit PARERR in SCS_STATUS should be tested as well.

Selection and reselection interrupts can be prevented by setting all the bits of SCS_SEL_ENA to 0.

3.9.5.2 DMA Count Reaches 0

An interrupt can be signalled when the DMA count register SCD_CNT reaches 0 during a DMA transfer. Such an interrupt occurs when the following conditions are met.

- The DMA bit in the SCS_MODE register is set.
- A DMA transfer to or from the disk buffer occurs, during which the count register SCD_CNT reaches 0.
- The INTEOP bit in the SCS_MODE register is set.

If the first two conditions are satisfied, the DMAEND bit in the SCS_STATUS register is set. If all three conditions are satisfied, INTREQ is also set in SCS_STATUS.

Note that when DMAEND is set, the system DMA controller performs no additional transfers, but the target's block transfer is not necessarily complete. When the system operates as an initiator, it must test the MATCH bit in SCS_STATUS to determine when the target has completed its data block. In addition, when sending data to the target, the system must monitor REQ in SCS_CUR_STAT and ACK in SCS_STATUS until both are false, to be sure that the last byte has been transferred.

3.9.5.3 Bus Parity Error

The 5380 chip can signal an interrupt when it detects invalid parity in data received from the SCSI tape bus. Such an interrupt is generated when the following conditions are met.

- The PARCK bit in the SCS_MODE register is set.
- Invalid parity is detected during a DMA transfer from the SCSI tape bus to the disk buffer, or during a processor read of the SCS_CUR_DATA register.
- The INTPAR bit in the SCS_MODE register is set.

If the first two conditions are satisfied, the PARERR bit in the SCS_STATUS register is set. If all three conditions are satisfied, INTREQ is also set in SCS_STATUS.

3.9.5.4 Phase Mismatch

The chip can signal an interrupt whenever the three bus phase signals MSG, C/D, and I/O do not match the corresponding bits in the SCS_TAR_CMD register during a DMA data transfer. The match state is continuously reflected in the MATCH bit of the SCS_STATUS register. The interrupt is signalled when all of the following conditions are met.

- The MATCH bit in SCS_STATUS is 0.
- The DMA bit in SCS_MODE is 1.
- The bus REQ signal is asserted to request a data transfer.

The identify status for such an interrupt is that DMA is 1 in SCS_MODE and that DMAEND and MATCH are both 0 in SCS_STATUS.

3.9.5.5 Bus Disconnect

The chip can generate an interrupt when the SCSI tape bus BSY signal becomes false. Such an interrupt is generated when the following conditions are met.

- The MONBSY bit in SCS_MODE is 1.
- The bus BSY signal goes false for at least 400 ns.

This condition sets the BSYERR bit in the SCS_STATUS register.

3.9.5.6 SCSI Tape Bus Reset

The chip generates an interrupt whenever the RST signal on the SCSI tape bus is asserted, either by another device on the bus or when the host program sets the RST bit in the SCS_INI_CMD register. When a reset occurs, the chip releases all bus signals within 800 ns. This interrupt cannot be disabled.

Note that the RST signal is not latched in the SCS_CUR_STAT register. So the RST bit may not still be set when the host responds to an interrupt which was caused by RST from another device on the bus.

3.9.6 Reset Conditions

The three possible reset conditions for the 5380 chip are described in the following paragraphs.

3.9.6.1 System Hardware Reset

At system power-on or when the system I/O reset signal is generated (Section 3.11.3), the 5380 chip is reinitialized and all internal logic and control registers are cleared. All signals are removed from the SCSI tape bus. This does not assert the RST signal on the SCSI tape bus.

3.9.6.2 RST Received from SCSI Tape Bus

When the RST signal is asserted on the bus by some other device, all the 5380 internal logic and registers are cleared, except that the interrupt request signal is asserted (bit INTREQ of SCS_STATUS) and the RST bit of the SCS_INI_CMD register is not altered.

3.9.6.3 RST Issued to SCSI Tape Bus

When the host program asserts RST on the SCSI tape bus by setting the RST bit in the SCS_INI_CMD register, all the 5380 internal logic and registers are cleared, except that the interrupt request signal is asserted (bit INTREQ of SCS_STATUS) and the RST bit of the SCS_INI_CMD register remains asserted until cleared by the host program or by a system hardware reset.

3.9.7 Programming Notes

This section contains hints that programmers should be aware of when writing drivers for the tape controller.

3.9.7.1 Using the Tape and Disk Controllers

The 5380 tape controller chip, the 9224 disk controller chip, and the disk data buffer share a common local data bus. This bus is used both by processer accesses to either chip or to the data buffer, and by DMA transfers between either chip and the data buffer. Therefore, it is not possible to use both the tape controller and the disk controller at the same time. Further, whenever either controller has an outstanding data transfer operation which will perform DMA transfers to or from the disk data buffer, the processor must not attempt to access the data buffer or any register in either controller chip until the chip signals that the current operation is done. Otherwise the processor access may collide with a DMA access cycle, which will corrupt the data transfer for all parties. One implication of this is that the interrupt system must be used by the controller chips to signal the completion of data transfer commands, since the processor cannot poll a controller chip during such a command.

3.9.7.2 Device ID Values

The SCSI device ID numbers shown in Table 3-39 are assigned by convention to the VS410 CPU on the system module and an attached TZK50 tape controller. For the CPU, this is done in its device driver code. The TZK50 tape controller requires that jumpers be set on its board, as described in the *TZK50/SCSI Controller Technical Manual.*

Table 3-39: Device ID Values

Device	ID number	ID bits
CPU	0	01 hex
TZK50	1	02 hex

In addition, both the CPU and TZK50 should assert and test parity on the SCSI tape bus. This is done for the CPU by asserting the PARCK bit in the SCS_MODE register. The TZK50 tape controller requires that a jumper be set on its module, as described in the *TZK50/SCSI Controller Technical Manual.*

3.10 ThinWire Ethernet Circuits

The only portion of the ThinWire Ethernet network circuitry that is not on the network option module is the transceiver circuitry. The transceiver circuitry is located in the upper right corner of the system module (see Figure 3-115). It consists of the coaxial cable connector, the coaxial transceiver interface chip, and the isolation transformer. Figure 3-116 shows a block diagram of the transceiver circuitry on the system module.

Figure 3-115: Transceiver Circuitry on System Module

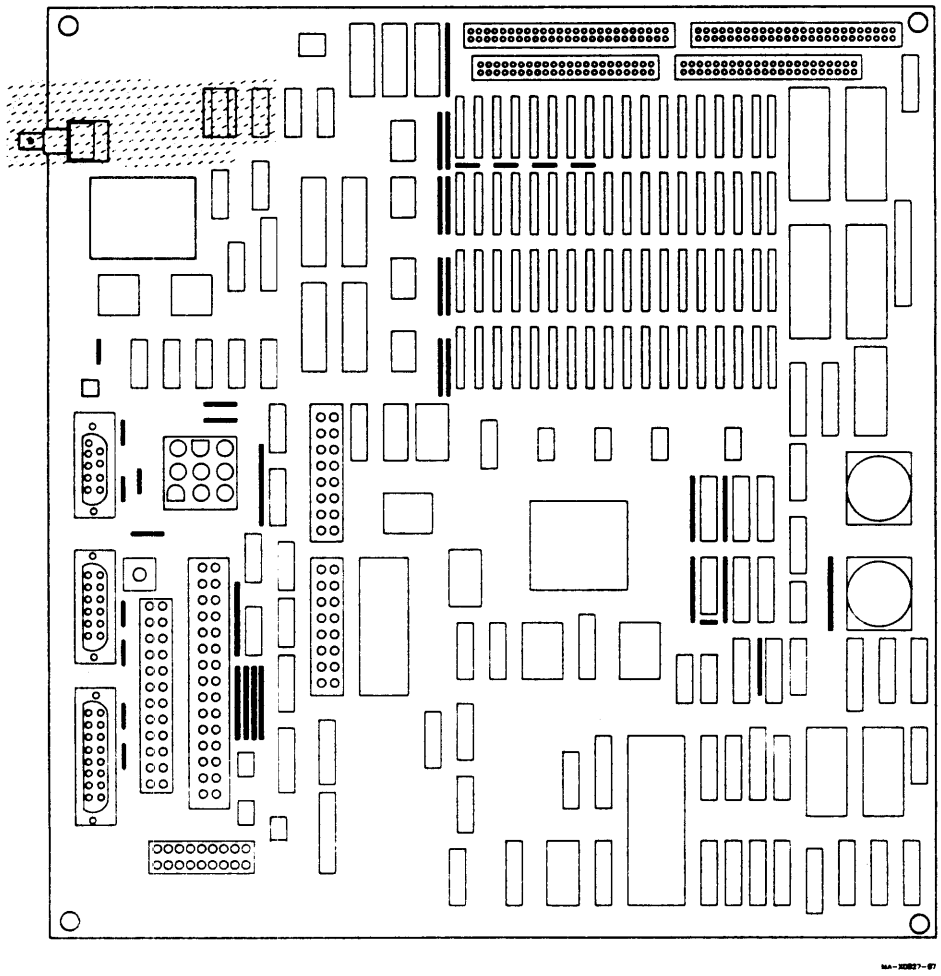
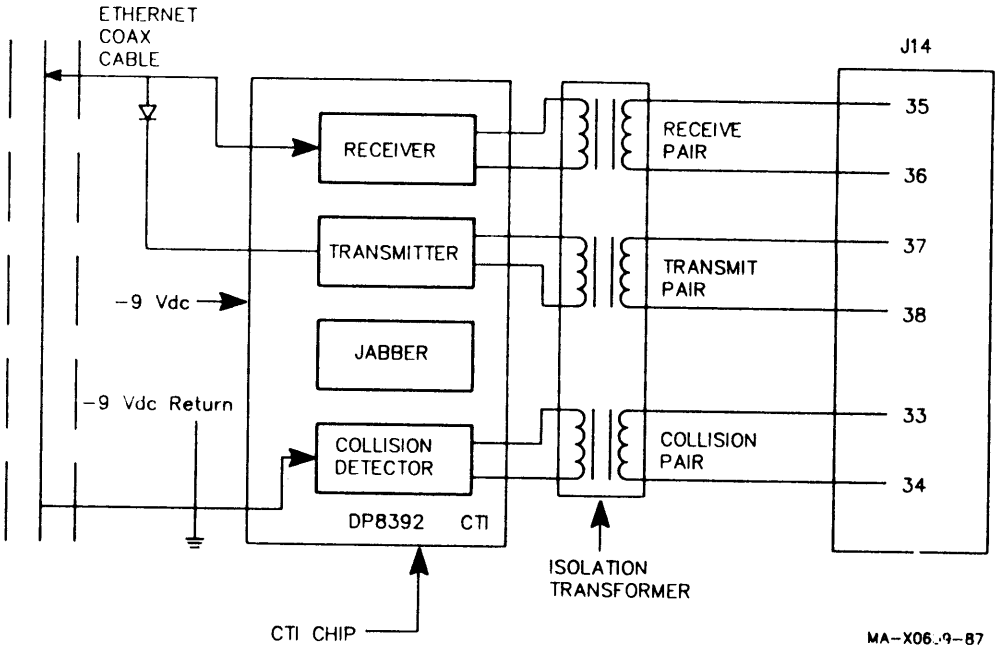


Figure 3-116: ThinWire Ethernet Transceiver Circuitry



3.10.1 Coaxial Transceiver Interface

The coaxial transceiver interface (CTI) is a DP8392 chip. It is used as the coaxial cable line driver and receiver for the ThinWire Ethernet local area network. The CTI contains a transmitter, receiver, collision detector, and a jabber timer. Figure 3-117 shows the DP8392 CTI chip and Table 3-40 lists a description of the pins.

Figure 3-117: Coaxial Transceiver Interface Chip Pinout

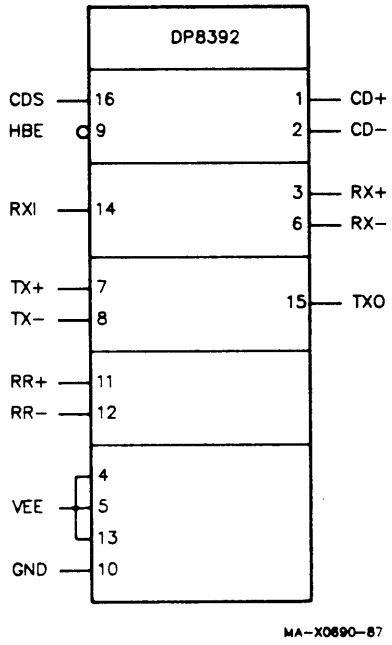


Table 3–40: Coaxial Transceiver Interface Chip Pinout

Pin	Signal	Description
1 2	CD+ CD-	These signals are the balanced differential line driver outputs from the collision detect circuitry.
3 6	RX+ RX-	These signals are the balanced differential line driver outputs from the receiver.
4,5,13	VEE	These signals are the power supply connections to the chip. VEE is -9 Vdc.
7 8	TX+ TX-	These signals are the balanced differential line receiver inputs to the transmitter.
9	HBE	This signal enables the collision detector heartbeat since it is asserted (grounded).
10	GND	This signal is the -9 Vdc return to the power supply.
11 12	RR+ RR-	These signals are connected to each other by a resistor to establish the operating currents within the chip.
14	RXI	This signal is the receive input from the coaxial cable.
15	TXO	This signal is the transmit output to the coaxial cable.
16	CDS	This signal is the ground sense connection for the collision detect circuit.

3.10.1.1 Transmitter

The transmitter section of the CTI consists of a differential line receiver and a current driver. The differential line receiver receives the transmit data from the network option module through an isolation transformer. The driver outputs the transmit data onto the coaxial cable.

3.10.1.2 Receiver

The receiver section of the CTI consists of four function blocks. They are the equalizer, a squelch circuit, an AC coupled comparator, and a differential line driver. The equalizer filters the incoming signal to compensate for the phase bias distortion from the coaxial cable. The squelch circuit prevents any noise on the coaxial cable from falsely triggering the receiver in the absence of the signal. The compensated signal is AC coupled to reduce slicing errors that can lead to a phase distortion. The output of the comparator then feeds to a differential line driver which sends the received data to the network option module through an isolation transformer.

3.10.1.3 Collision Detector

The collision detection circuitry consists of a low pass filter, a voltage reference, a 10 MHz oscillator and a differential line driver. The low pass filter is used to determine the DC voltage level of the signal on the coaxial cable. When a collision occurs, the output of the filter exceeds the reference voltage, and a 10 MHz oscillator collision signal is generated. The signal first passes onto the line driver and then through an isolation transformer before it arrives at the network option module.

3.10.1.4 Jabber

The jabber circuitry functions as a watchdog timer to terminate longer than legal length data packets by disabling the transmitter. The collision signal is then asserted to indicate this condition to the network option module. When the network module terminates the transmission, the jabber is automatically reset after a time delay. The jabber is also reset at power-up.

3.10.2 Network Address ROM

A 32-byte ROM on the system module contains a unique network address for each system. The physical address of each system is determined at the time of manufacture. Data from this ROM is read in the low-order bytes of 32 consecutive longwords at physical addresses 2009.0000 through 2009.007C. The network address occupies the first six bytes (addresses 2009.0000 through 2009.0014). The byte at 2009.0000 is the first byte to be transmitted or received in an address field of an Ethernet packet. Its low-order bit (bit 0) is transmitted or received first in the serial bit stream. This ROM is installed in a socket so it can be removed from a failing system module and installed on the new system module.

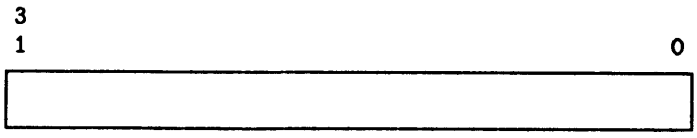
3.11 Miscellaneous System Registers

This section describes three miscellaneous system registers and one address strobe delay line.

3.11.1 HALT Code Register (HLTCOD)

The halt code register (HLTCOD) is a read/write longword register at physical address 2008.0000. It is intended for use by the ROM-resident firmware program which handles a processor restart. This program moves internal processor register SAVISP to HLTCOD so that the restart code can be extracted without accessing any of the processor's general registers or any RAM locations. Figure 3-118 shows the halt code register.

Figure 3-118: Halt Code Register (HLTCOD)



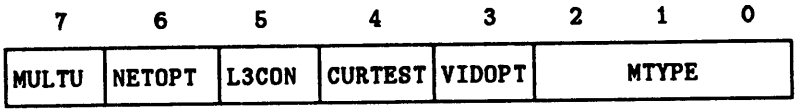
NOTE: *There is an interaction between the HLTCOD register and the SCD_CNT register (described in Section 3.9.4.2). The contents of HLTCOD must be zero whenever a program attempts to read the contents of SCD_CNT; otherwise the value received may be in error. The contents of HLTCOD do not affect program writes to SCD_CNT and do not affect actual DMA operation.*

3.11.2 Configuration and Test Register (CFGTST)

The configuration and test register (CFGTST) is a read-only byte register at physical address 2002.0000. This register is a tri-state octal driver that stores such information as whether this system is a VAXstation 2000 or a MicroVAX 2000, and which option slots contain option modules. This register is enabled by the SYSREGEN L signal from the standard cell. When enabled, the system configuration information is driven on to the BDAL07:00 bus. Figure 3-119 shows the configuration and test register.

NOTE: *The CFGTST register shares its physical address with the IORESET register (Section 3.11.3). Programs must not be designed to write to the CFGTST register, since this will generate an I/O reset signal.*

Figure 3-119: Configuration and Test Register (CFGTST)



Data Bit	Definition
MULTU	Multi-char user (bit 7). This bit is set by jumper W6 on the system module. It is a 1 when the system module is used in a MicroVAX 2000 system. This bit is a 0 when the system module is used in a VAXstation 2000 system.
NETOPT	Network option present (bit 6). This bit is 1 when a board is present in the network option module connector.
L3CON	Line 3 console (bit 5). This bit is 1 when pins 8 and 9 of the printer connector are connected together by the BCC08 console cable. This bit is 0 when pins 8 and 9 are not connected together (BCC05 printer cable or no cable is connected).
CURTEST	Cursor test (bit 4). This bit is the complement of the Test pin output from the monochrome video cursor chip.
VIDOPT	Video option present (bit 3). This bit is 1 when a module is present in the video option module connector.
MTYPE	Memory option type (bits 2:0). These bits indicate the size of memory option module (if any) inserted into the memory option module connector. The values of the data bits are listed below.

Value	Definition
000	No board present
001	1024 Kbytes
010	2048 Kbytes
011	4096 Kbytes
100	Reserved
101	Reserved
110	Reserved
111	Reserved

3.11.3 I/O Reset Register (IORESET)

The I/O reset register (IORESET) is a write-only byte register at physical address 2002.0000. Any write access to this register generates a reset signal to the following four controllers (the data contained in this register is ignored).

- 9224 disk controller chip (Section 3.8)
- 5380 SCSI bus controller chip (Section 3.9)
- Network controller option (Chapter 5)
- The controller installed in the general purpose option port.

NOTE: *Consult the individual sections for details of the effects of writing to IORESET. Also note that the CPU, FPU, interrupt controller, and serial line controller are not affected by the IORESET.*

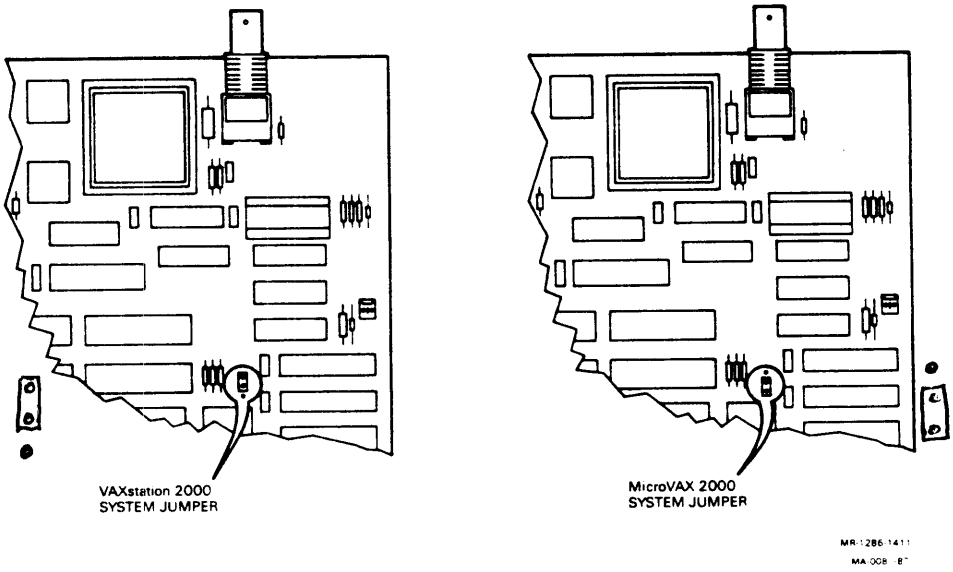
3.11.4 Address Strobe Delay Line

Address strobe timing for the memory chips due to the bus delays may not be present when needed by these circuits. The address strobe delay circuit holds the address strobe for 50 ns after the first clock pulse of CLKO is received following the assertion of AS. The product of this delay circuit is called a buffered address strobe (BAS1). This allows BAS1 to stay asserted 50 ns after AS is deasserted.

3.12 System Jumper Configuration

The system module for the VAXstation 2000 and MicroVAX 2000 systems are identical. The only way for the system to know whether it is a VAXstation 2000 or a MicroVAX 2000 is by the position of the system jumper. The system jumper sets a bit in the configuration and test register. Figure 3-120 shows the system jumper setting.

Figure 3–120: VAXstation 2000 and MicroVAX 2000 System Jumper



3.13 System Module Connector Pinouts

The following tables list the signals on each connector on the system module.

Table 3–41: Power Connector (J1)

Pin	Signal	Pin	Signal	Pin	Signal
1	Ground	2	Ground	3	Ground
4	-12 Vdc	5	+5 Vdc	6	+5 Vdc
7	+12 Vdc	8	-9 Vdc return	9	-9 Vdc

Table 3–42: ThinWire Ethernet Connector (J2)

Pin	Signal	Pin	Signal
1	Outer shell (ground)	2	Center conductor

Table 3–43: Printer Connector (J3)

Pin	Signal	Pin	Signal	Pin	Signal
1	Chassis ground	2	PTR_XDAT	3	PTR_RDATA
4	No connection	5	+12 Vdc	6	No connection
7	Chassis ground	8	Ground	9	FER_ENA

Table 3–44: Battery Connector (J4)

Pin	Signal	Pin	Signal
1	Plus side of battery	2	Negative side of battery

Table 3–45: Video Connector (J5)

Pin	Signal	Pin	Signal	Pin	Signal
1	VID_RED	2	Color return	3	Monochrome return
4	Fused +5 Vdc	5	AUX_RDAT	6	Keyboard ground
7	Chassis ground	8	Fused +12 Vdc	9	Monochrome signal
10	VID_GREEN	11	VID_BLUE	12	-12 Vdc
13	AUX_XDAT	14	KBD_RDAT	15	KBD_XDAT

Table 3–46: Network Option Module Connector (J6)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Ground	2	Ground	3	BDAL31	4	BDAL30
5	BDAL29	6	BDAL28	7	BDAL27	8	BDAL26
9	BDAL25	10	BDAL24	11	BDAL23	12	BDAL22
13	Ground	14	Ground	15	BDAL21	16	BDAL20
17	BDAL19	18	BDAL18	19	BDAL17	20	BDAL16
21	BDAL15	22	BDAL14	23	BDAL13	24	BDAL12
25	BDAL11	26	BDAL10	27	Ground	28	Ground
29	BDAL09	30	BDAL08	31	BDAL07	32	BDAL06
33	BDAL05	34	BDAL04	35	BDAL03	36	BDAL02
37	BDAL01	38	BDAL00	39	Ground	40	Ground

Table 3–47: RD/RX Connector (J7)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Ground	2	LOSPEED	3	Ground	4	RXINDEX
5	Ground	6	RXSEL0	7	Ground	8	No conn.
9	Ground	10	MORON	11	Ground	12	RXDIR
13	Ground	14	RXSTEP	15	Ground	16	RXWD
17	Ground	18	RXWRGT	19	Ground	20	RXTK00
21	Ground	22	WRTPROT	23	Ground	24	RXRDATA
25	Ground	26	RXHSEL0	27	Ground	28	RXRDY
29	RDHSEL3	30	RDHSEL2	31	Ground	32	RDWRGT
33	SKCOMPL	34	Ground	35	RDTK00	36	WRTFAULT
37	Ground	38	RDHSEL0	39	RDHSEL1	40	Ground
41	RDINDEX	42	RDRDY	43	Ground	44	RDSTEP
45	RDSEL0	46	Ground	47	RDSEL1	48	Ground
49	RDDIR	50	DSELACK	51	Ground	52	No conn.
53	No conn.	54	Ground	55	RD0_WDATH	56	RD0_WDATL

Table 3–47 (Cont.): RD/RX Connector (J7)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
57	Ground	58	RD0_RDATH	59	RD0_RDATL	60	Ground

Table 3–48: Graphics Option Port Connector (J8)

Pin	Signal	Pin	Signal	Pin	Signal
1	+5 Vdc	2	+5 Vdc	3	+12 Vdc
4	-12 Vdc	5	Ground	6	BCLKO
7	BRESET	8	BAS1	9	VDS
10	BWRITE1	11	VDBE	12	MEMAD0
13	Ground	14	Ground	15	CAS3
16	CAS2	17	CAS1	18	CAS0
19	NIIRQ1	20	NIIRQ2	21	MEMAD7
22	REFCYC	23	OPTROMENA	24	OPTVIDENA
25	OPTIRQ	26	OPTEOF	27	Ground
28	Ground	29	INTENA	30	SCYC/IAD2
31	DCYC/IAD1	32	STFH/IAD0	33	VID_RED
34	Ground	35	VID_GREEN	36	Ground
37	VID_BLUE	38	Ground	39	OPTPRESENT
40	+5 Vdc				

Table 3–49: Expansion Disk Read/Write Cable Connector (J9)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	DSELACK	2	Ground	3	No conn.	4	Ground
5	No conn.	6	Ground	7	+5 Vdc	8	Ground
9	No conn.	10	No conn.	11	Ground	12	Ground
13	RD1_WDATH	14	RD1_WDATL	15	Ground	16	Ground
17	RD1_RDATH	18	RD1_RDATL	19	Ground	20	No conn.

Table 3–50: Communication Connector (J10)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	No conn.	2	COM_XDAT	3	COM_RDAT	4	COM_RTS
5	COM_CTS	6	COM_DSR	7	Ground	8	COM_CAR
9	No conn.	10	No conn.	11	No conn.	12	COM_SPDMI
13	No conn.	14	No conn.	15	No conn.	16	No conn.
17	No conn.	18	LLP_BCK	19	No conn.	20	COM_DTR
21	No conn.	22	COM_RI	23	COM_DSRS	24	No conn.
25	COM_TMI						

Table 3–51: Graphics Option Port Connector (J11)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Ground	2	Ground	3	BDAL31	4	BDAL30
5	BDAL29	6	BDAL28	7	BDAL27	8	BDAL26
9	BDAL25	10	BDAL24	11	BDAL23	12	BDAL22
13	Ground	14	Ground	15	BDAL21	16	BDAL20
17	BDAL19	18	BDAL18	19	BDAL17	20	BDAL16
21	BDAL15	22	BDAL14	23	BDAL13	24	BDAL12
25	BDAL11	26	BDAL10	27	Ground	28	Ground
29	BDAL09	30	BDAL08	31	BDAL07	32	BDAL06
33	BDAL05	34	BDAL04	35	BDAL03	36	BDAL02
37	BDAL01	38	BDAL00	39	Ground	40	Ground

Table 3–52: Memory Option Module Connector (J12)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Ground	2	Ground	3	BDAL31	4	BDAL30
5	BDAL29	6	BDAL28	7	BDAL27	8	BDAL26
9	BDAL25	10	BDAL24	11	BDAL23	12	BDAL22
13	Ground	14	Ground	15	BDAL21	16	BDAL20
17	BDAL19	18	BDAL18	19	BDAL17	20	BDAL16
21	BDAL15	22	BDAL14	23	BDAL13	24	BDAL12
25	BDAL11	26	BDAL10	27	Ground	28	Ground
29	BDAL09	30	BDAL08	31	BDAL07	32	BDAL06
33	BDAL05	34	BDAL04	35	BDAL03	36	BDAL02
37	BDAL01	38	BDAL00	39	Ground	40	Ground

Table 3–53: Tape Port Connector (J13)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Ground	2	DBUS0	3	Ground	4	DBUS1
5	Ground	6	DBUS2	7	Ground	8	DBUS3
9	Ground	10	DBUS4	11	Ground	12	DBUS5
13	Ground	14	DBUS6	15	Ground	16	DBUS7
17	Ground	18	DBUSP	19	Ground	20	Ground
21	Ground	22	Ground	23	Ground	24	Ground
25	No conn.	26	No conn.	27	Ground	28	Ground
29	Ground	30	Ground	31	Ground	32	SCATN
33	Ground	34	Ground	35	Ground	36	SCBSY
37	Ground	38	SCACK	39	Ground	40	SCRST
41	Ground	42	SCMSG	43	Ground	44	SCSEL
45	Ground	46	SCC/D	47	Ground	48	SCREQ
49	Ground	50	SCI/O				

Table 3–54: Network Option Module Connector (J14)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	+5 Vdc	2	+5 Vdc	3	+12 Vdc	4	-12 Vdc
5	Ground	6	BCLK0	7	BRESET	8	VAS
9	VDS	10	VWRITE	11	VDBE	12	MEMAD0
13	Ground	14	Ground	15	CAS3	16	CAS2
17	CAS1	18	CAS0	19	VBM3	20	VBM2
21	VBM1	22	VBM0	23	NIROMCS	24	NIENA
25	NIIRQ1	26	NIIRQ2	27	REFCYC	28	VDMG
29	DMAREQ	30	SCYC/IAD2	31	PERROR	32	READY
33	CD+	34	CD-	35	RX+	36	RX-
37	TX+	38	Tx-	39	NIPRESENT	40	+5 VDC

Table 3–55: Memory Option Module Connector (J15)

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	+5 VDC	2	+5 VDC	3	Ground	4	Ground
5	PBIT03	6	PBIT02	7	PBIT01	8	PBIT00
9	MSIZE2	10	MEMAD8	11	MEMAD7	12	MEMAD6
13	Ground	14	Ground	15	MEMAD5	16	MEMAD4
17	MEMAD3	18	MEMAD2	19	MEMAD1	20	MEMAD0
21	MSIZE1	22	MSIZE0	23	CAS3	24	CAS2
25	CAS1	26	CAS0	27	Ground	28	Ground
29	BDAL22	30	ERAS	31	SRAS0	32	BDAL21
33	BDAL20	34	BAS1	35	VDBE	36	BWRITE
37	Ground	38	Ground	39	+5 Vdc	40	+5 Vdc

3.14 Power Requirements

The system module requires +12 Vdc, +5 Vdc, and -12 Vdc supplies for operation and a special -9 Vdc for power loading at 180 milliamps for the ThinWire Ethernet transceiver circuits on the system module.

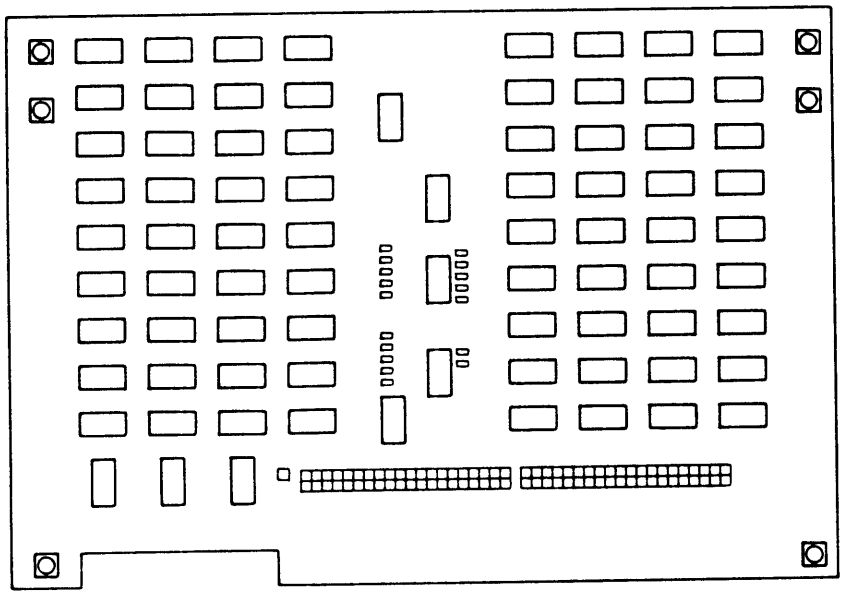
Chapter 4
MS400 Option Memory Modules

4.1 Introduction

This chapter describes the MS400-AA and MS400-BA memory modules that are options to the KA410-AA system module. These modules do not provide RAM control signal generation; however, they do provide transceivers for data and buffers for driving the RAM array with RAS, CAS, WRITE, and ADDRESS. The KA410-AA system module generates byte parity when writing to RAM memory and checks byte parity when reading from RAM memory. Parity checking applies both to CPU accesses and to DMA accesses generated by the network controller option. Only those bytes selected by the processor byte mask are affected and checked.

The MS400-AA memory module contains 2 megabytes of memory and the MS400-BA memory module contains 4 megabytes of memory. The MS400-BA has components on both sides of the module. Only one MS400 memory module may be connected to a KA410-AA system module. Figure 4-1 shows a front view of the MS400 memory module (note that the MS400-BA has components on both sides).

Figure 4-1: MS400 Memory Module



MAX-0002-87

4.2 Theory of Operation

MS400 option memory is contained in DRAMs. These are the same DRAMs as described in Section 3.3.1.1. The control signals on the memory module and the timing cycles are described in this section.

4.2.1 Memory Module Control Signal Descriptions

Signal ERAS L is the RAS timing signal for the memory on the option module. ERAS is asserted for normal read and write cycles on the memory module (such as physical addresses in the range 0020.0000 through 00FF.FFFF). Signal SRAS L is the RAS timing signal for RAM memory on the base system module (physical addresses in the range 0000.0000 through 001F.FFFF). SRAS is negated during normal read and write cycles on the memory module. During refresh cycles, both ERAS and SRAS are asserted.

Bits 22, 21 and 20 of the system data/address bus (BDAL22, BDAL21, and BDAL20 on the system module that map to MSEL22, MSEL21, and MSEL20,respectively on the memory module) are latched in an F373 latch on the falling edge of VAS L. These latched address bits are decoded by an F138 which generates RAS for one of the four (or two) 1-megabyte memory arrays on the module. The appropriate decoder output is gated by ERAS true and SRAS false during normal read and write cycles and is input to the DRAM chip's RAS pins.

During a refresh cycle, both ERAS and SRAS are asserted. This negates all the outputs of the decoders and switches the multiplexers to assert RAS to all the DRAM chips on the option module.

The four CASx L signals from the system module pass through F244 buffers and series damping resistors to the CAS pins on the DRAM chips. Each CAS signal is associated with one of the processor byte masks and so determines which bytes of a longword are affected by a memory read or write cycle.

The multiplexed address lines MEMADDx H from the system module pass through F244 buffers and series damping resistors to the address pins on the DRAM chips. The timing of row address, RAS assertion, column address, and CAS assertion are controlled by the system module.

Signal BWRITE L from the system module passes through F244 buffers to the WE pins on the DRAM chips. This signal also controls the signal flow direction in the F245 data transceivers.

The data input (D) and output (Q) pins of each DRAM chip are wired together and are sent to the system module data/address bus through F245 transceivers. The transceivers are enabled when both ERAS L and VDBE L are asserted. The direction of data flow is selected by the BWRITE L signal.

4.2.2 Memory Cycles

The memory module responds to three types of memory cycles. They are the read, write, and refresh cycles. Each cycle on the module is initiated by the assertion of ERAS L. The cycle type is determined by SRAS L and BWRITE L as shown in Table 4-1. The timing cycles for the memory module are described in Section 3.5.2.

Table 4–1: Determining Memory Cycles

Cycle Type	ERAS L	SRAS L	BWRITE L
Read	True	False	False
Write	True	False	True
Refresh	True	True	False

4.3 Connector Pinouts

Connector J1 carries power, address, and control signals as listed in Table 4–2. Connector J2 carries the buffered processor data/address bus (BDAL31:00) as listed in Table 4–3.

Table 4–2: Connector J1 Pinout

Pin	Signal	Description
1	+5 VC	
2	+5 VB	
3	GND	
4	GND	
5	PBIT03 H	Parity bit for byte 3
6	PBIT02 H	Parity bit for byte 2
7	PBIT01 H	Parity bit for byte 1
8	PBIT00 H	Parity bit for byte 0
9	MSIZE2 L	Memory size bit 2
10	MEMAD8 H	Multiplexed address bit 8
11	MEMAD7 H	Multiplexed address bit 7
12	MEMAD6 H	Multiplexed address bit 6
13	GND	
14	GND	
15	MEMAD5 H	Multiplexed address bit 5
16	MEMAD4 H	Multiplexed address bit 4

Table 4-2 (Cont.): Connector J1 Pinout

Pin	Signal	Description
17	MEMAD3 H	Multiplexed address bit 3
18	MEMAD2 H	Multiplexed address bit 2
19	MEMAD1 H	Multiplexed address bit 1
20	MEMAD0 H	Multiplexed address bit 0
21	MSIZE1 L	Memory size bit 1
22	MSIZE0 L	Memory size bit 0
23	CAS3 L	CAS for byte 3
24	CAS2 L	CAS for byte 2
25	CAS1 L	CAS for byte 1
26	CAS0 L	CAS for byte 0
27	GND	
28	GND	
29	MSELC H	BDAL22 H from system
30	ERAS L	Extended RAS (ERAS from the standard cell)
31	SRAS L	Standard RAS (SRAS0 from the standard cell)
32	MSELB H	BDAL21 H from system
33	MSELA H	BDAL20 H from system
34	VAS L	Address strobe (BAS1 L on system module)
35	VDBE L	Data bus enable
36	BWRITE L	Write (BWRITE1 L on system module)
37	GND	
38	GND	
39	+5 VA	
40	+5 VA	

Table 4-3: Connector J2 Pinout

Pin	Signal	Pin	Signal
1	GND	21	BDAL15 H
2	GND	22	BDAL14 H
3	BDAL31 H	23	BDAL13 H
4	BDAL30 H	24	BDAL12 H
5	BDAL29 H	25	BDAL11 H
6	BDAL28 H	26	BDAL10 H
7	BDAL27 H	27	GND
8	BDAL26 H	28	GND
9	BDAL25 H	29	BDAL09 H
10	BDAL24 H	30	BDAL08 H
11	BDAL23 H	31	BDAL07 H
12	BDAL22 H	32	BDAL06 H
13	GND	33	BDAL05 H
14	GND	34	BDAL04 H
15	BDAL21 H	35	BDAL03 H
16	BDAL20 H	36	BDAL02 H
17	BDAL19 H	37	BDAL01 H
18	BDAL18 H	38	BDAL00 H
19	BDAL17 H	39	GND
20	BDAL16 H	40	GND

4.4 Configuration Jumpers

There are no field-modifiable jumpers on the module. The version of the module is determined by three signals on connector J1. The first two are the same for both memory modules but the third signal is either disconnected or grounded to indicate which memory module is installed. Table 4-4 lists the three signals and the preset configuration jumpers for both memory modules.

Table 4-4: Memory Module Configuration Jumpers

Signal	Pin	MS400-AA	MS400-BA
MSIZE2 L	9	Open	Open
MSIZE1 L	21	Ground	Ground
MSIZE0 L	22	Open	Ground

4.5 Power Requirements

The memory modules require + 5 volts DC with a tolerance of plus or minus five percent. The typical current drawn is .5 amps.

Chapter 5

ThinWire Ethernet (DESVA) Option Module

5.1 Introduction

The DESVA Ethernet controller option module enables the connection of a VAXstation 2000 or MicroVAX 2000 system to an Ethernet network via a ThinWire connection using RG-58 coaxial cable. The option is packaged on a 4-inch by 7-inch board that is located in the system unit and plugs into the two DESVA option connectors (J5 and J14) on the system module. The DESVA module is powered by the system box power supply. The DESVA contains a Local Area Network Controller for Ethernet (LANCE) chip, a serial interface adapter (SIA) chip, and a ROM that contains device-driver programming, and supports logic circuitry. The Ethernet transceiver chip, Ethernet address ROM, and the BNC connector for the RG-58 cable to the Ethernet are mounted on the system module. The network components on the system module are inactive until the DESVA option module is installed.

5.2 Connector Pin Descriptions

Figure 5-1 is a diagram of the Network Interconnect module. Table 5-1 and Table 5-2 show the pin assignments for connectors J1 and J2.

Figure 5-1: Network Interconnect Module

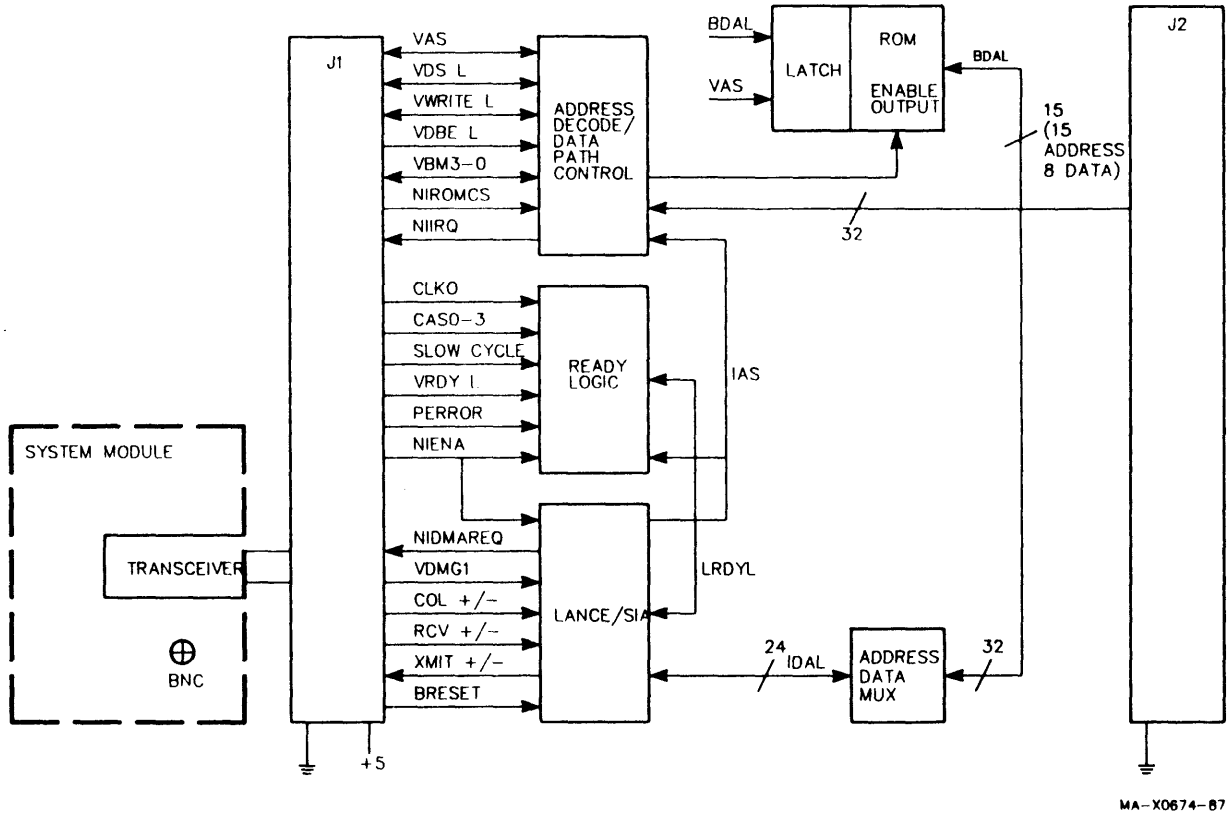


Table 5-1: Pin Assignments for Connector J1

Pin Number	Signal Name	Description
1	-	+5 V
2	-	+5 V
3	-	Not used (0 V)
4	-	Not used (0 V)
5	-	Not used (0 V)
6	VCLK0	Clock out from CPU. When LANCE is DMA master, LANCE waits for 3 VCLK0 cycles before next memory transfer.
7	BRESET	Buffered reset from CPU
8	VAS	Address strobe from CPU
9	VDS	Data strobe from CPU
10	VWRITE	Write from CPU
11	VDBE	Data buffer enable from CPU
12	-	Not used (0 V)
13	-	Not used (0 V)
14	-	Not used (0 V)
15	CAS3	Address strobe from standard cell, generated when LANCE is DMA master. These signals are generated in response to byte mask and address strobe signals from the LANCE as an acknowledgement that memory timing has been started.
16	CAS2	Address strobe from standard cell, generated when LANCE is DMA master. These signals are generated in response to byte mask and address strobe signals from the LANCE as an acknowledgement that memory timing has been started.
17	CAS1	"
18	CAS0	"
19	VBM3	Byte mask. Generated during DMA using LANCE byte mask signals, and sent to standard cell so that appropriate CAS signals can be generated.

Table 5–1 (Cont.): Pin Assignments for Connector J1

Pin Number	Signal Name	Description
20	VBM2	Byte mask. Generated during DMA using LANCE byte mask signals, and sent to standard cell so that appropriated CAS signals can be generated.
21	VBM1	"
22	VBM0	"
23	NIROMCS	DESVA ROM chip select from standard cell
24	NIENA	DESVA enable to LANCE chip select from standard cell
25	NIIRQ1	DESVA interrupt request
26	-	Not used (0 V)
27	-	Not used (0 V)
28	VDMG	DMA grant from CPU
29	DMAREQ	DMA request from DESVA
30	SLOW CYCLE	When LANCE is DMA slave, after NIENA is asserted SLOW CYCLE is then asserted to cycle-slip the CPU while writing to CSRs.
31	PERROR	Parity error - inhibits DMA transfer.
32	VRDY L	Ready from CPU. Bidirectional: input when LANCE is DMA master, output when LANCE is DMA slave. As an input, VRDY L tells LANCE to proceed with DMA transfer.
33	COLL+	Collision detect from transceiver on system module to serial interface adapter (SIA)
34	COLL-	Collision detect from transceiver on system module to serial interface adapter (SIA)
35	RECV+	Receive + from transceiver on system module to SIA
36	RECV-	Receive - from transceiver on system module to SIA
37	XMIT+	Transmit + from SIA to transceiver on system module
38	XMIT-	Transmit - from SIA to transceiver on system module

Table 5–1 (Cont.): Pin Assignments for Connector J1

Pin Number	Signal Name	Description
39	NIPRESENT	DESVA module present
40	-	+5 V

Table 5–2: Pin Assignments for Connector J2

Pin Number	Signal Name	Description
1	-	Not used (0 V)
2	-	Not used (0 V)
3	BDAL 31	Bus data and address line
4	BDAL 30	"
5	BDAL 29	"
6	BDAL 28	Bus data and address line
7	BDAL 27	Bus data and address line
8	BDAL 26	"
9	BDAL 25	"
10	BDAL 24	"
11	BDAL 23	"
12	BDAL 22	"
13	-	Not used (0 V)
14	-	Not used (0 V)
15	BDAL 21	Bus data and address line
16	BDAL 20	"
17	BDAL 19	"
18	BDAL 18	"
19	BDAL 17	"
20	BDAL 16	Bus data and address line
21	BDAL 15	"
22	BDAL 14	"

Table 5–2 (Cont.): Pin Assignments for Connector J2

Pin Number	Signal Name	Description
23	BDAL 13	"
24	BDAL 12	"
25	BDAL 11	"
26	BDAL 10	"
27	-	Not used (0 V)
28	-	Not used (0 V)
29	BDAL 09	Bus data and address line
30	BDAL 08	"
31	BDAL 07	"
32	BDAL 06	"
33	BDAL 05	"
34	BDAL 04	Bus data and address line
35	BDAL 03	"
36	BDAL 02	"
37	BDAL 01	"
38	BDAL 00	"
39	-	Not used (0 V)
40	-	Not used (0 V)

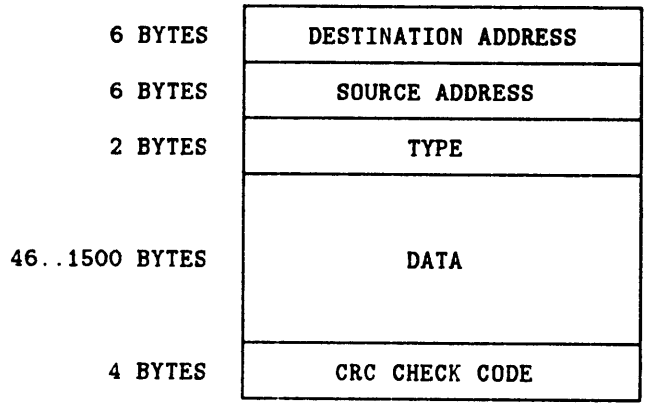
5.3 Ethernet Implementation

This option module supports the physical link and data link layers of the Ethernet protocol.

5.3.1 Packet Format

Data is passed over the Ethernet at a serial data rate of 10 million bits per second in variable-length packets. Figure 5-2 shows the format of each packet.

Figure 5-2: Ethernet Packet Format



The minimum size of a packet in this implementation is 64 bytes, which implies a minimum data length of 46 bytes. Packets shorter than this are called "runt packets" and are treated as erroneous when received by the network controller.

5.3.2 Network Addresses

There are two types of network addresses. Both are 48 bits (6 bytes) long.

1. Physical address: The unique address associated with a particular station on an Ethernet, which should be distinct from the physical address of any other station on any other Ethernet.
2. Multicast address: A multi-destination address associated with one or more stations on a given Ethernet (sometimes called a logical address). There are two kinds of multicast addresses:

a. Multicast-group address: An address associated by higher-level convention with a group of logically related stations.

b. Broadcast address: A predefined multicast address which denotes the set of all the stations on the Ethernet.

Bit 0 (the least significant bit of the first byte) of an address denotes the type: it is 0 for physical addresses and 1 for multicast addresses. In either case the remaining 47 bits form the address value. A value of 48 ones is always treated as the broadcast address.

The physical address of each VAXstation 2000 or MicroVAX 2000 system is determined at the time of manufacture and is stored in the Ethernet Address ROM on the main system board (see Section 3.3).

5.4 LANCE Chip Overview

This section describes the LANCE chip.

5.4.1 LANCE Description

The LANCE is a 10-megabits per second MOS device in a 48-pin package that implements the Ethernet network access algorithm. The LANCE performs direct-memory access (DMA), error reporting, and packet handling. In addition, the LANCE listens for a clear coaxial cable before transmitting, and handles collisions.

The LANCE chip is a microprogrammed controller that can conduct extensive operations independently of the MicroVAX CPU. There are four control and status registers (CSRs) within the LANCE chip that are programmed by the the MicroVAX CPU chip to initialize the LANCE chip and start its independent operation. Once started, the LANCE uses its built-in DMA controller to directly access RAM memory to get additional operating parameters and to manage the buffers it uses to transfer packets to and from the Ethernet. The LANCE uses three structures in memory.

1. Initialization block—24 bytes of contiguous memory starting on a word boundary. The initialization block is set up by the central processor and is read by the LANCE when the processor starts the Lance's initialization process. The initialization block contains the system's network address and pointers to the receive and transmit descriptor rings; it is described in Section 5.10 below.

2. Descriptor rings— two logically circular rings of buffer descriptors, one ring used by the chip receiver for incoming data and one ring used by the chip transmitter for outgoing data. Each buffer descriptor in a ring is 8 bytes long and starts on a quadword boundary. It points to a data

buffer elsewhere in memory, contains the size of that buffer, and holds various status information about the buffer's contents.

3. Data buffers—contiguous portions of memory to buffer incoming or outgoing packets. Data buffers must be at least 64 bytes long (100 bytes for the first buffer of a packet to be transmitted) and may begin on any byte boundary.

When the MicroVAX or VAXstation system is ready to begin network operation, the central processor sets up the initialization block, the receive descriptor ring, the transmit descriptor ring, and each of their data buffers in memory. The central processor then starts the LANCE by writing to its CSRs. The LANCE performs its initialization process and then enters its polling loop. In this loop, the LANCE listens to the network for packets whose destination addresses it recognizes. It also scans the transmit descriptor ring for descriptors that have been marked by the CPU to indicate that they contain outgoing data packets. When the LANCE detects a recognizable network packet, it receives and stores that packet in one or more receive buffers and marks their descriptors accordingly. When the LANCE finds a packet to be transmitted, it transmits it to the network and marks its descriptor when transmission is complete. Whenever the LANCE completes a reception or transmission (or encounters an error condition), it sets flags in its control and status register 0 to signal the CPU (usually by an interrupt) that it has done something important.

5.4.2 Transmit Mode

In transmit mode, the LANCE chip directly accesses data in a transmit buffer in memory. The LANCE prefaces the data with a preamble and a sync pattern, and calculates and appends a 32-bit CRC. This packet is then ready for serial transmission to the SIA. On transmission, the first byte of data loads into the 48-byte FIFO. The LANCE then begins to transmit a preamble while simultaneously loading the rest of the the packet into FIFO for transmission.

5.4.3 Receive Mode

In receive mode, packets are sent via the SIA to the LANCE. The packets are loaded into the 48-byte FIFO for preparation of automatic downloading into buffer memory. A CRC is calculated and compared with the CRC appended to the data packet. If the calculated CRC checksum doesn't agree with the packet CRC, an error bit is set and an interrupt is generated to the CPU.

5.4.4 LANCE Chip Pinout

Figure 5-3 and Table 5-3 describe the LANCE chip pinout.

Figure 5-3: LANCE Chip Pinout

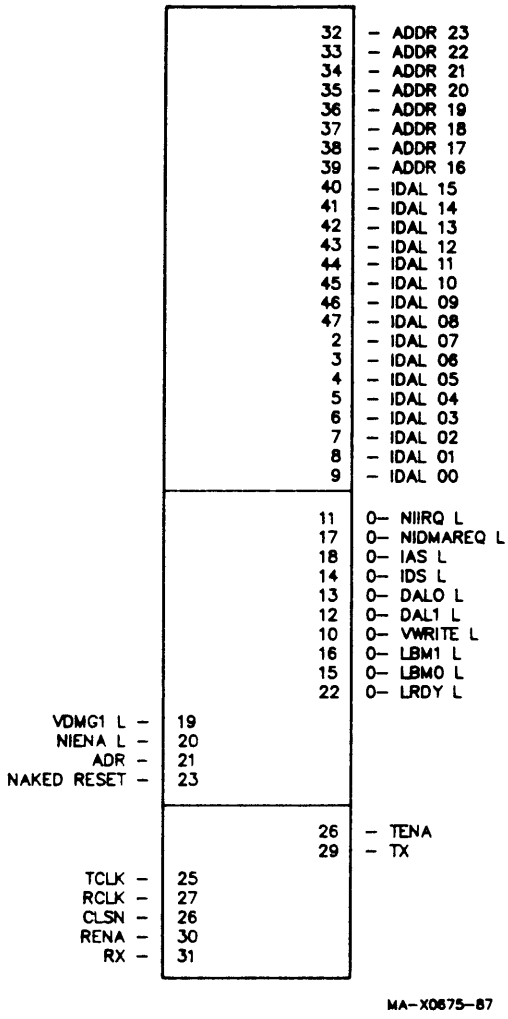


Table 5–3: LANCE Chip Pin Descriptions

Pin	Description
IDAL00 - IDAL15	Data/address lines (input/output tri-state). The time multiplexed address/date bus. During the address portion of a memory transfer, DAL00 - DAL15 contain the lower 16 bits of the memory address. The upper 8 bits of the address are contained in A16 - A23.
ADDR16 - ADDR23	High order address bus (output tri-state). The additional address bits necessary to extend the DAL lines to access a 24-bit address. These lines are driven by a bus master only.
VWRITE L	(Input/Output tri-state). Indicates the type of operation to be performed in the current bus cycle. This signal is an output when the LANCE is a bus master. High - Data is taken off the DAL by the chip. Low - Data is placed on the DAL by the chip. VWRITE L is an input when the LANCE is a bus slave. High - Data is taken off the DAL by the chip. High - Data is taken off the DAL by the chip.
LBM1 L, LBM0 L	(Output tri-state). Pins 15 and 16 are programmable through bit 00 of CSR3. If CSR3 bit 00 BCON = 0, pin 15 = BM0 L (output tri-state) and pin 16 = LBM1 L (output tri-state). LBM0 L, LBM1 L(byte mask). This indicates the byte(s) on the DAL are to be read or written during this bus transaction. The LANCE drives these lines only as a bus master. It ignores the byte mask lines when it is a bus slave, and assumes word transfers. The following lines describe byte selection using byte mask:

LBM1 L	LBM0 L	
Low	Low	Whole word
Low	High	Upper byte
High	Low	Lower byte
High	High	None

Table 5-3 (Cont.): LANCE Chip Pin Descriptions

Pin	Description
	If CSR3 bit 00 BCON = 1, pin 15 = BYTE (output tri-state) and pin 16 = BUSAKO L (output) Byte selection may also be done using the BYTE line and DAL00 line, latched during the address portion of the bus cycle. The LANCE drives BYTE only as a bus master and ignores it when a bus slave selection is done (similar to LBM0 L, LBM1 L). Byte selection is described as follows:

Byte	DAL00	
Low	Low	Whole word
Low	High	Upper byte
High	Low	Lower byte
High	High	None

	BUSAKO L is a bus request daisy chain output. If the chip is not requesting the bus and it receives VDMG1 L, BUSAKO L is driven low. If the LANCE is requesting the bus when it receives VDMG1 L, BUSAKO L remains high.
NIENA L	Chip select (input). When asserted, this signal indicates that the LANCE is the slave device of the data transfer. NIENA L must be valid throughout the data portion of the bus cycle. NIENA L must not be asserted when VDMG1 L is low.
ADR	Register address port select (input). When LANCE is a slave, ADR indicates which of the two register ports is selected. ADR LOW selects register data port. ADR HIGH selects register address port. ADR must be valid throughout the data portion of the bus cycle and is used only by the LANCE when NIENA L is low.
IAS L	Address latch enable/Address enable (output tri-state). Used to demultiplex the DAL lines and define the address portion of the bus cycle. This I/O pin is programmable through bit 01 of the CSR3.

Table 5–3 (Cont.): LANCE Chip Pin Descriptions

Pin	Description
	As Address Latch Enable (CSR3 bit 01, ACON = 0), the signal pulses low during the address portion of the transfer and remains low during the data portion. ALE can be used by a slave device to control a latch on the bus address lines. When ALE is high the latch is open and when ALE goes low the latch is closed.
	AS address enable (CSR3 bit 01, ACON = 1), the signal pulses low during the address portion of the bus transaction. The low to high transition of AS can be used by a slave device to strobe the address into a register.
	The LANCE drives the IAS L line only as a bus master.
IDS	Data strobe (input/output tri-state). Defines the data portion of the bus transaction. IDS is high during the address portion of a bus transaction and low during the data portion. The low to high transition can be used by a slave device to strobe bus data into a register. DAS L is driven only as a bus master.
DALO L	Data/Address line out (output tri-state). An external bus transceiver control line. DALO L is asserted when the LANCE drives the DAL lines. DALO L is low only during the address portion if the transfer is a READ. It is low for the entire transfer if the transfer is a WRITE. DALO L is driven only when the LANCE is a bus master.
DALI L	Data/Address line in (output tri-state). An external bus transceiver control line. DALI L is asserted when the LANCE reads from the DAL lines. It is low during the data portion of a READ transfer and remains high for the entire transfer if it is a WRITE. DALI L is driven only when LANCE is a bus master.
NIDMAREQ L	Bus hold request (output open drain). Asserted by the LANCE when it requires access to memory. NIDMAREQ L is held low for the entire ensuing bus transaction. The function of this pin is programmed through bit 00 of CSR3. Bit 00 of CSR3 is cleared when NAKED RESET L is asserted.
VDMG1 L	Bus hold acknowledge (input). A response to NIDMAREQ L. When VDMG1 L is low in response to the chip's assertion of NIDMAREQ L, the chip is the bus master. VDMG1 L deasserts upon the deassertion of NIDMAREQ L.

Table 5–3 (Cont.): LANCE Chip Pin Descriptions

Pin	Description
NIIRQ L	Interrupt (output open drain). An attention signal that indicates, when active, that one or more of the following CSR0 status flags is set: BABL, MERR, MISS, RINT, TINT, or IDON. NIIRQ L is enabled by bit 06 of CSR0 (INEA = 1). NIIRQ L is asserted until the source of the interrupt is removed.
RX	Receive (output). Receive input bit stream.
TX	Transmit (output). Transmit output bit stream.
TENA	Transmit Enable (output). Transmit output bit stream enable. A level asserted with the transmit output bit stream, TX, to enable the external transmit logic.
RCLK	Receive clock (input). A 10 MHz square wave synchronized to the receive data and only active while receiving an input bit stream.
CLSN	Collision (input). A logical input that indicates that a collision is occurring on the channel.
RENA	Receive enable (input). A logical input that indicates the presence of carrier on the channel.
TCLK	Transmit clock (input). 10 MHz clock.
LRDY L	(Input/Output open drain). When the LANCE is a bus master, LRDY L is an asynchronous acknowledgement from the bus memory that LANCE can accept data in a WRITE cycle or that is has put data on the DAL lines in a READ cycle. As a bus slave, the LANCE asserts LRDY L when it has put data on the DAL lines during a READ cycle, or is about to take data off the DAL lines during a WRITE cycle. LRDY L is a response to IDS and returns high after IDS has gone high. LRDY L is an output when the LANCE is a bus master and an output when the LANCE is a bus slave.
NAKED RESET L	Bus Request Signal (input). Causes the LANCE to cease operation, clear its internal logic, and enter an idle state with the stop bit of CSR0 set.
Vcc	Power supply pin. +5 volts (+/- 5 %)
Vss	Ground.

5.5 SIA Chip Overview

Figure 5-4 shows the pinout for the serial interface adapter (SIA) chip.

Figure 5-4: SIA Chip Pinout

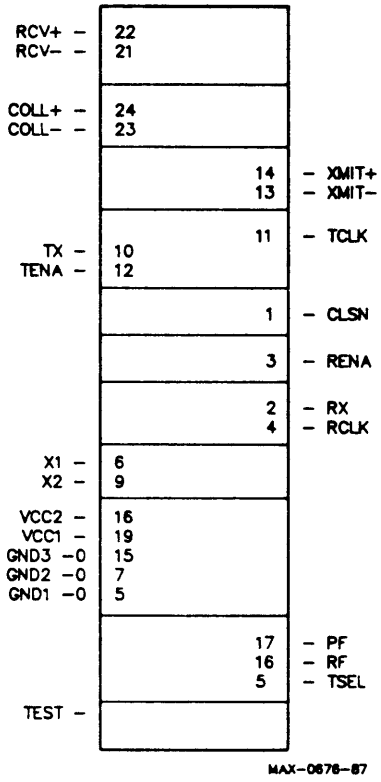


Table 5–4: SIA Chip Pin Descriptions

Pin Name	Description
CLSN	Collision (output). A TTL active high output. Signals at the collision +/- terminals meeting threshold and pulse width requirements produce a logic high at CLSN output. When no signal is present at Collision +/-, CLSN output is low.
RX	Receive data (output). A MOS/TTL output, recovered data. When there is no signal at Receive +/-, and TEST L is high, RX is high. RX is activated with RCLK and remains active until end of message. During reception RX is synchronous with RCLK and changes after the rising edge of RCLK. When TEST L is low, RX is enabled.
RENA	Receive enable (output). TTL active high output. When there is no signal at Receive +/-, and TEST L is high, RENA is low. Signals meeting threshold and pulse width requirements produce a logic high at RENA. When Receive +/- becomes idle, RENA returns to the low state synchronous with the rising edge of RCLK.
RCLK	Receive clock (output). A MOS/TTL output recovered clock. When there is no signal at Receive +/-, and TEST L is high, RCLK is low. RCLK is activated after the third negative data transition at Receive +/-, and remains active until end of message. When TEST L is low, RCLK is enabled.
TX	Transmit (output). TTL compatible input. When TENA is high, signals at TX meeting setup and hold time to TCLK is encoded as normal Manchester at Transmit+ and Transmit-. TX high: TRANSMIT+ is negative with respect to Transmit- for first half of data bit cell. TX low: Transmit+ is positive with respect to Transmit- for first half of data bit cell.
TENA	Transmit enable (input). TTL compatible input. Active high data encoder enable. Signals meeting setup and hold time to TCLK allow encoding of Manchester data from TX to Transmit+ and Transmit-.
TCLK	Transmit clock (output). MOS/TTL output. TCLK provides symmetrical high and low clock signals at data rate for reference timing of data to be encoded. It also provides clock signals for the LANCE chip and an internal timing reference for receive path voltage-controlled oscillators.

Table 5–4 (Cont.): SIA Chip Pin Descriptions

Pin Name	Description
XMIT+/XMIT-	Transmit (outputs). A differential line output. This line pair is intended to operate into terminated transmission lines. For signals meeting setup and hold time to TCLK at TENA and TX, Manchester clock and data are output at Transmit+ and Transmit-.
RCV+/RCV–	Receiver (inputs). A differential input. A pair of internally biased line receivers consisting of a carrier detect receiver with offset threshold and noise filtering to detect the signal, and a data recovery receiver with no offset for Manchester data decoding.
COL+/COL-	Collision (a differential input). An internally biased line receiver input with offset threshold and noise filtering. Signals at COL +/- have no effect on data path functions.
TSEL	Transmit mode select. An open collector output and sense amplifier input. TSEL low: Idle transmit state. TRANSMIT+ is positive with respect to TRANSMIT-. TSEL high: Idle transmit state. TRANSMIT+ and TRANSMIT- are equal, providing "zero" differential to operate transformer coupled loads. When connected with an RC network, TSEL is held low during transmission. At the end of transmission the open collector output is disabled, allowing TSEL to rise and provide a smooth transition from logic high to "zero" differential idle. Delay and output return to "zero" are externally controlled by the RC time constant TSEL.
X1,X2	Biased crystal oscillator. X1 is the input and X2 is the bypass port. When connected for crystal operation, the system clock which appears at TCLK is half the frequency of the crystal oscillator. X1 may be driven from an external source of two times the data rate.
RF	Frequency setting voltage-controlled oscillator (VCO) loop filter. This loop filter output is a reference voltage for the receive path phase detector. It also is a reference for timing noise immunity circuits in the collision and receive enable path. Nominal reference VCO gain is 1.25 TCLK frequency MHz/V.
PF	Receive path VCO phase-lock loop filter. This loop filter input is the control for receive path loop damping. Frequency of the receive VCO is internally limited to transmit frequency +/- 12%. Nominal receive VCO gain is 0.25 reference VCO gain MHz/V.

Table 5–4 (Cont.): SIA Chip Pin Descriptions

Pin Name	Description
TEST L	Test control (input). A static input that is connected to Vcc for normal SIA operation and to ground for testing of receive path function. When TEST L is grounded, RCLK and RX are enabled so that receive path loop may be functionally tested.
GND1	High current ground
GND2	Logic ground
GND3	Voltage-controlled oscillator ground
Vcc1	High current and logic supply
Vcc2	Voltage-controlled oscillator supply

5.5.1 SIA Description

The SIA has three basic functions. It is a Manchester encoder/line driver in the transmit path, a Manchester encoder with noise filtering and lock-on characteristics in the recieve path, and a signal detect/converter in the collision path. The SIA provides the interface between the TTL logic environment of the LANCE and the differential signalling environment in the transceiver cable.

5.5.2 Transmit Mode

The Manchester encoder in the SIA takes transmitted data from the LANCE and creates the Manchester-encoded differential signals TRANSMIT+ and TRANSMIT- to drive the transceiver cable. These differential signals are coupled through the transceiver (on the system module) and on to the Ethernet coaxial cable.

5.5.3 Receive Mode

When a carrier signal is present on the Ethernet coaxial cable, the transceiver creates the differential signals RECEIVE+ and RECEIVE-. These inputs to the SIA are decoded by the Manchester decoder. A phase-locked loop in the SIA synchronizes to the Ethernet preamble, allowing the decoder to recover clock and data from the cable, indicating to the LANCE that receive data and clock are available.

5.6 DMA Operation

The LANCE chip contains a built-in DMA controller that can transfer data directly between the chip and system memory in the address range 0000.0000 through 00FF.FFFF. (Only system module RAM and option module RAM appear in this address range.) The LANCE contains a 48-byte FIFO buffer to allow for DMA service latency and to minimize the number of request-grant arbitration cycles. When transferring large amounts of data in burst mode, the chip transfers 16 bytes per DMA request. Each longword transfer requires 0.6 microseconds, so a 16-byte burst requires either 2.4 or 3.0 microseconds, depending upon whether or not the data block is longword-aligned.

The LANCE's DMA controller is used to read the initialization block, to read and write the descriptor rings, and to read and write data buffers. Note that all the memory addresses handled by the chip are physical addresses. Programs which operate with CPU memory management enabled must translate their addresses from virtual to physical form before presenting them to the LANCE chip.

If the (parity enable) PEN bit of the system's memory system error register (MSER) is set, then parity is checked during DMA read cycles. When a parity error is detected, the ERR signal is asserted as described in Section 3.3. Such an error manifests itself in one of two ways: If another DMA cycle immediately follows the DMA cycle during which the error occurred (that is, during the same DMA request), then the MERR bit of the NI_CSR0 register is set but no CPU machine check occurs. If, however, the DMA cycle during which the error occurred was followed by a CPU cycle (that is, the failing DMA cycle was the last in a DMA request), then a machine check occurs, but MERR is not set. In both cases, the PER bit of MSER is set and the address of the failing location is latched in the MEAR register.

5.7 Controller Firmware ROM

Figure 5-5 shows the pinout for the controller firmware ROM and Table 5-5 describes the pins.

Figure 5-5: Controller Firmware ROM

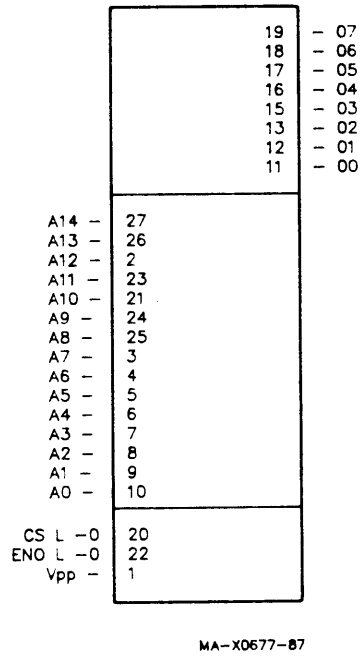


Table 5-5: ROM Pin Descriptions

Pin Name	Description
00 - 07	Data outputs to memory
A0 - A14	Address inputs, latched in from DAL bus by VAS L signal from CPU
Chip select L	Enables the data path to DAL bus, along with VWRITE L and VDS L signals from CPU
Enable Output L	Output always enabled

5.7.1 ROM Description

The network controller option board contains one 28-pin socket for a ROM to contain option identification information and device driver programming. This ROM contains 32 kilobytes and is connected to the low-order 8 bits of the system data bus. Therefore, its contents appear as the low-order byte in each of 32-kilobyte consecutive longwords in the address range 2010.0000 through 2011.FFFF (the data returned in the three high-order bytes of each longword is unpredictable). See Section 3.3.2.3 for information on address allocation and ROM format.

If the option-present signal is asserted, the ROM is checked and its contents unloaded into memory. TEST 1 code is then executed. Since the ROM is connected only to the low-order byte of the data bus, code cannot be directly executed from the ROM; it must be copied into consecutive bytes of a RAM area and executed from there.

5.8 Program Control of the LANCE

Program control of the LANCE chip is via two 16-bit read/write ports, each of which appears as the low-order word of a longword address. These ports are:

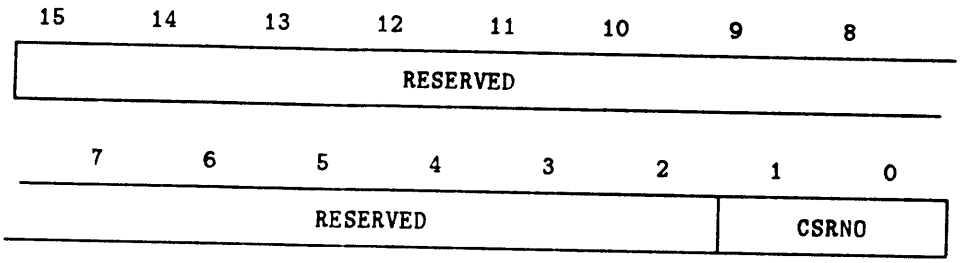
Address	Name	Description
200E.0000	NI_RDP	Register data port
200E.0004	NI_RAP	Register address port

These ports provide access to four 16-bit control and status registers which are named NI_CSR0 through NI_CSR3. A CSR is accessed by first writing its number into the register address port NI_RAP after which the contents of the CSR are read or written by accesses to the register data port NI_RDP. Note that registers other than NI_CSR0 may be accessed only while the STOP bit of NI_CSR0 is set.

5.8.1 Register Address Port (NI_RAP)

The register address port is a 16-bit read/write port at physical address 200E.0004. It selects which of the four CSR's is accessed via the register data port. Figure 5-6 shows the LANCE register address port format.

Figure 5-6: LANCE Register Address Port (NI_RAP) Format



Bit	Definition
<15:2>	Reserved. Ignored on write; read as 0's.
CSRNO	CSR select (bits 1:0). These read/write bits select which of the four CSRs is accessible via the register data port. They are cleared to 0 at power-on. Values are as follows:

Bits 1:0	Register
0 0	NI_CSR0
0 1	NI_CSR1
1 0	NI_CSR2
1 1	NI_CSR3

5.8.2 Register Data Port (NI_RDP)

The register data port at physical address 200E.0000 is a 16-bit window through which the CPU can read and write the CSR designated by the register address port NI_RAP.

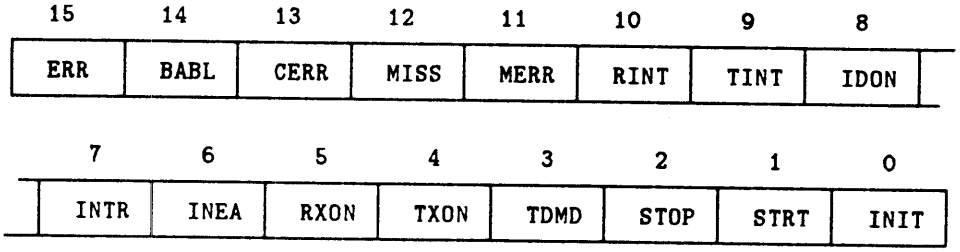
Note that registers NI_CSR1, NI_CSR2, and NI_CSR3 are accessible only while the STOP bit in NI_CSR0 is set. If that STOP bit is clear (that is, the LANCE chip is active), attempts to read from those CSR's return UNDEFINED data and attempts to write to them are ignored. Accesses to a CSR via NI_RDP do not alter the register address pointer NI_RAP. In normal operation, only NI_CSR0 can be accessed, so NI_RAP should be set to point to NI_CSR0 and left that way.

5.8.3 Control and Status Register 0 (NI_CSR0)

This register is used by the controlling program to start and stop the operation of the LANCE chip and to monitor its status. It is accessible to the processor via port NI_RDP when bits 1:0 of NI_RAP are set to 00. All of its bits can be read at any time and none of its bits is affected by reading the register. The effects of a write operation are described individually for each bit.

When power is applied to the system, all the bits in this register are cleared except the STOP bit, which is set. Figure 5-7 shows the LANCE control and status register.

Figure 5-7: LANCE Control and Status Register 0 (NI_CSR0)



Bit	Definition
ERR	Error summary (bit 15). This read-only bit is 1 whenever any of of the bits BABL, CERR, MISS, or MERR in this register are 1's. Writing to this bit has no effect. It is cleared when all of the bits which set it are 0 or when the STOP bit is set.
BABL	Transmitter timeout error (bit 14). This bit is set when the transmitter has been on the channel longer than the time required to send the maximum length packet. It is set after 1519 data bytes have been transmitted (the chip continues to transmit until the whole packet is transmitted or until a failure occurs before the whole packet is transmitted). This bit is cleared when a 1 is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the ERR and INTR bits are also 1's.
CERR	Collision error (bit 13). This bit is set when the collision input to the chip failed to activate within 2 microseconds after a chip-initiated transmission is completed. This collision-after-transmission is a transceiver test feature. This function is also known as heartbeat or SQE (signal quality error) test. This bit is cleared when a 1 is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the ERR bit is also 1.
MISS	Missed packet (bit 12). This bit is set when the receiver loses a packet because it does not own a receive buffer. The MISS bit is not valid in internal loopback mode. This bit is cleared when a one is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the ERR and INTR bits are also 1's.
MERR	Memory error (bit 11). This bit is set when the chip attempts a DMA transfer and does not receive a ready response from the memory within 25.6 microseconds after beginning the memory cycle. This condition occurs when a parity error occurred on an immediately preceding DMA bus read cycle that asserted the ERR signal. When MERR is set, the receiver and transmitter are turned off (bits RXON and TXON of this register are cleared to 0). This bit is cleared when a 1 is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the ERR and INTR bits are also 1's.
RINT	Receive interrupt (bit 10). This bit is set when the chip updates an entry in the receive descriptor ring for the last buffer received or when reception is stopped due to a failure.

Bit	Definition
	This bit is cleared when a 1 is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the INTR bit is also 1.
TINT	Transmitter interrupt (bit 9). This bit is set when the chip updates an entry in the transmit descriptor ring for the last buffer sent or when transmission is stopped due to a failure.
	This bit is cleared when a 1 is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the INTR bit is also 1.
IDON	Initialization done (bit 8). This bit is set when the chip completes the initialization process which was started by setting the INIT bit in this register. When IDON is set, the chip has read the initialization block from memory and stored the new parameters.
	This bit is cleared when a 1 is written to it (writing a 0 has no effect) or when the STOP bit is set. When this bit is 1, the INTR bit is also 1.
INTR	Interrupt request (bit 7). This read-only bit is 1 whenever any of the bits BABL, MISS, MERR, RINT, TINT, or IDON in this register are 1's. Writing to this bit has no effect. It is cleared when all of the bits which set it are 0 or when the STOP bit is set.
	When both the INTR and INEA bits in this register are set, an interrupt request is sent to the system interrupt controller.
INEA	Interrupt enable (bit 6). This read/write bit controls whether the setting of the INTR bit generates an interrupt request. When both the INTR and INEA bits in this register are set, an interrupt request is sent to the system interrupt controller.
	This bit is set when a 1 is written to it. It is cleared when a 0 is written to it or when the STOP bit is set.
RXON	Receiver on (bit 5). This read-only bit, when set to 1, indicates that the receiver is enabled. RXON is set when initialization is completed (that is, when IDON is set, unless the DRX bit of the initialization block MODE register is 1) and then the STRT bit in this register is set. Writing to this bit has no effect. RXON is cleared when either the MERR or STOP bits of this register are set.
TXON	Transmitter on (bit 4). This read-only bit, when set to 1, indicates that the transmitter is enabled. TXON is set when initialization is completed (that is, when IDON is set, unless the DTX bit of the initialization block MODE register is 1) and then the STRT bit in this register is set. Writing to this bit has no effect. TXON is cleared when either the MERR or STOP bits of this register are set or when any of bits UFLO, BUFF, or RTRY in a transmit buffer descriptor are set.

Bit	Definition
TDMD	Transmit demand (bit 3). Setting this bit signals the chip to access the transmit descriptor ring without waiting for the polltime interval to elapse. This bit need not be set to transmit a packet; setting it merely hastens the chip's response to the insertion of a transmit descriptor ring entry by the host program. This bit is set by writing a 1 to it (writing a 0 has no effect) and is cleared by the chip when it recognizes the bit (the bit may read as 1 for a short time after it is set, depending upon the level of activity in the chip). TDMD is also cleared when the STOP bit is set.
STOP	Stop external activity (bit 2). Setting this bit stops all external activity and clears the internal logic of the chip; this has the same effect as the electrical reset signalled at power-on. The chip remains inactive and STOP remains set until the STRT or INIT bits in this register are set. This bit is set by writing a 1 to it (writing a 0 has no effect) or at power-on. It is cleared when either INIT or STRT is set. If the processor writes 1's to STOP, INIT, and STRT at the same time, STOP takes precedence and neither STRT nor INIT is set. Setting STOP clears all the other bits in this register. After STOP has been set, the other three CSRs (NI_CSR1, NI_CSR2, and NI_CSR3) must be reloaded before setting INIT or STRT (note that those three registers may be accessed only while STOP is set).
STRT	Start operation (bit 1). Setting this bit enables the chip to send and receive packets, perform DMA and manage the buffer. The STOP bit must be set prior to setting the STRT bit (setting STRT then clears STOP). STRT is set by writing a 1 to it (writing a 0 has no effect). It is cleared when the STOP bit is set.
INIT	Initialize (bit 0). Setting this bit causes the chip to perform its initialization process, which reads the initialization block from the memory addressed by the contents of NI_CSR1 and NI_CSR2 using DMA accesses. The STOP bit must be set prior to setting the INIT bit (setting INIT then clears STOP). INIT is set by writing a 1 to it (writing a 0 has no effect). It is cleared when the STOP bit is set.

NOTE: *The INIT and STRT bits must not be set at the same time.*

The proper initialization procedure is as follows:

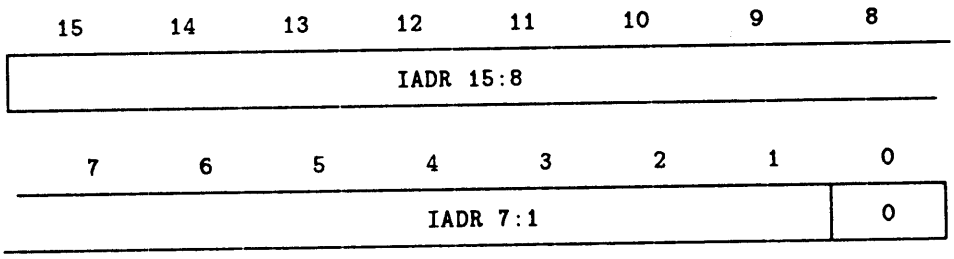
1. Set STOP in NI_CSR0.
2. Set up the initialization block in memory.

3. Load NI_CSR1 and NI_CSR2 with the starting address of the initialization block.
4. Set INIT in NI_CSR0.
5. Wait for IDON in NI_CSR0 to become set.
6. Set STRT in NI_CSR0 to begin the operation.

5.8.4 Control and Status Register 1 (NI_CSR1)

This read/write register is used in conjunction with NI_CSR2 to supply the 24-bit physical memory address of the initialization block, which the chip reads when it performs its initialization process. The register is accessible to the processor via NI_RDP when bits 1:0 of NI_RAP are 01 and the STOP bit of NI_CSR0 is set. Its contents at power-on are unpredictable. Figure 5-8 shows the LANCE control and status register 1.

Figure 5-8: LANCE Control and Status Register 1 (NI_CSR1)

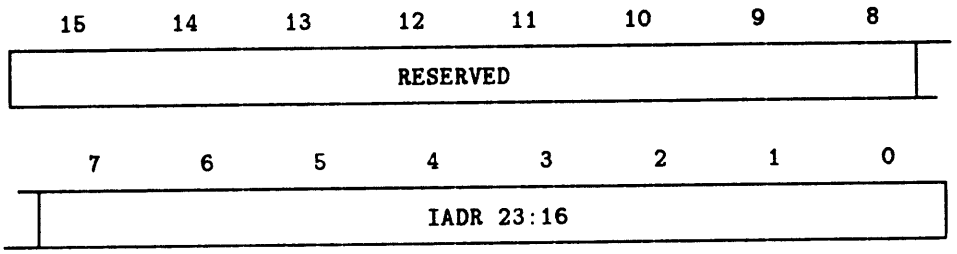


Bit	Definition
IADR	Initialization block address (bits 15:0). These are the low-order sixteen bits of the (24-bit physical) byte address of the first byte of the initialization block. Because the block must be word-aligned, bit 0 must be zero.

5.8.5 Control and Status Register 2 (NI_CSR2)

This read/write register is used in conjunction with NI_CSR1 to supply the 24-bit physical memory address of the initialization block which the chip reads when it performs its initialization process. The register is accessible to the processor via NI_RDP when bits 1:0 of NI_RAP are 10 and the STOP bit of NI_CSR0 is set. Its contents at power-on are unpredictable. Figure 5-9 shows the LANCE control and status register 2.

Figure 5-9: LANCE Control and Status Register 2 (NI_CSR2)

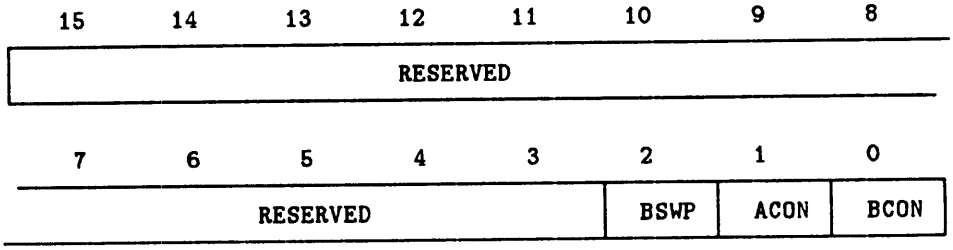


Bit	Definition
<15:8>	Reserved. Write with 0's.
IADR	Initialization block address (bits 7:0). These are the high-order eight bits of the (24-bit physical) byte address of the first byte of the initialization block.

5.8.6 Control and Status Register 3 (NI_CSR3)

This read/write register controls certain aspects of the electrical interface between the LANCE chip and the system. It must be set as indicated for each bit. The register is accessible to the processor via NI_RDP when bits 1:0 of NI_RAP are 11 and the STOP bit of NI_CSR0 is set. Its contents at power-on are entirely 0's. Figure 5-10 shows the LANCE control and status register 3.

Figure 5-10: LANCE Control and Status Register 3 (NI_CSR3)



Bit	Definition
<15:3>	Reserved. Ignored on write; read as 0's.
BSWP	Byte swap (bit 2). When this bit is set, the chip will swap the high and low bytes for DMA data transfers between the silo and bus memory in order to accommodate processors which consider bus bits 15:08 to be the least significant byte of data. This bit is read/write; it is cleared when the STOP bit in NI_CSRO is set. For this system, this bit must be 0.
ACON	ALE control (bit 1). This bit controls the polarity of the signal emitted on the chip's ALE/AS pin during DMA operation. This bit is read/write; it is cleared when the STOP bit in NI_CSRO is set. For this system this bit must be 0.
BCON	Byte control (bit 0). This bit controls the configuration of the byte mask and hold signals on the chip's pins during DMA operation. This bit is read/write; it is cleared when the STOP bit in NI_CSRO is set. For this system, this bit must be 0.

5.9 Interrupts

The LANCE chip asserts an interrupt request signal whenever the INTR and INEA bits in NI_CSR0 are both 1's. This signal is presented to the system interrupt controller as interrupt number 5, the "network controller primary" source. Its vector number is 250 hexadecimal.

The change of the interrupt signal from false to true sets bit NP in the interrupt request register (INT_REQ), which generates a CPU interrupt when the corresponding bit in the interrupt mask register INT_MSK is also set. Note that since the input to INT_REG is transition sensitive rather than level sensitive, a program which services an interrupt request from the LANCE must either service all the conditions which contributed to the setting of the INTR bit in NI_CSR0 so that INTR becomes 0, or must generate another transition

of the interrupt signal by setting the INEA bit of NI_CSR0 to 0 and then back to 1 again. Interrupt number 4, the "network controller secondary" source, is not used by this option.)

5.10 Initialization Block

When the LANCE chip is initialized (by setting the INIT bit in NI_CSR0), it reads a 24-byte block of data called the initialization block from main memory using DMA accesses. The physical address of the initialization block (IADR) is taken from NI_CSR1 and NI_CSR2. Since the data must be word-aligned, the low-order bit of the address must be 0. The initialization block comprises twelve 16-bit words arranged as shown in Figure 5-11.

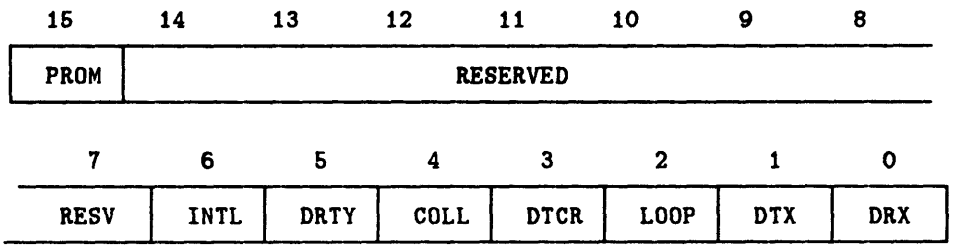
Figure 5-11: LANCE Initialization Block Format

IADR + 0	MODE	
IADR + 2	PADR <15:00>	
IADR + 4	PADR <31:16>	
IADR + 6	PADR <47:32>	
IADR + 8	LADRF <15:00>	
IADR + 10	LADRF <31:16>	
IADR + 12	LADRF <47:32>	
IADR + 14	LADRF <63:48>	
IADR + 16	RDRA <15:00>	
IADR + 18	RLEN	RDRA <23:16>
ADR + 20	TDRA <15:00>	
ADR + 22	TLEN	TDRA <23:16>

5.10.1 Initialization Block MODE Word (NIB_MODE)

The mode word of the initialization block allows alteration of the LANCE chip's normal operation for testing and special applications. For normal operation the mode word is entirely 0. Figure 5-12 shows the initialization block mode word.

Figure 5-12: Initialization Block Mode Word (NIB_MODE)



Bit	Definition
PROM	Promiscuous mode (bit 15). When this bit is set, all incoming packets are accepted regardless of their destination addresses.
<14:7>	Reserved. Should be written with 0's.
INTL	Internal loopback (bit 6). This bit is used in conjunction with the LOOP bit in this word to control loopback operation. See the description of the LOOP bit within this figure.
DRTY	Disable retry (bit 5). When this bit is set, the chip attempts only one transmission of a packet. If there is a collision on the first transmission attempt, a retry error (RTRY) is reported in the transmit buffer descriptor.
COLL	Force collision (bit 4). Setting this bit allows the collision logic to be tested. The chip must be in internal loopback mode for COLL to be used. When COLL is 1 a collision is forced during the subsequent transmission attempt. This results in 16 total transmission attempts with a retry error reported in NI_TMD3.
DTCR	Disable transmit CRC (bit 3). When DTCR is 0 the transmitter generates and appends a 4-byte CRC to each transmitted packet(normal operation). When DTCR is 1, the CRC logic is allocated instead to the receiver and no CRC is sent with a transmitted packet.

Bit	Definition
	During loopback, setting DTCR to 0 causes a CRC to be generated and sent with the transmitted packet, but no CRC check can be done by the receiver since the CRC logic is shared and cannot both generate and check a CRC at the same time. The CRC transmitted with the packet is received and written into memory following the data where it can be checked by software. If DTCR is set to 1 during loopback, the driving software must compute and append a CRC value to the data to be transmitted. The receiver checks this CRC upon reception and report any error.
LOOP	Loopback control (bit 2). Loopback allows the LANCE chip to operate in full duplex mode for test purposes. The maximum packet size is limited to 32 data bytes (in addition to which 4 CRC bytes may be appended). During loopback, the runt packet filter is disabled because the maximum packet is forced to be smaller than the minimum size Ethernet packet (64 bytes). Setting LOOP to 1 allows simultaneous transmission and reception for a packet constrained to fit within the silo. The chip waits until the entire packet is in the silo before beginning serial transmission. The incoming data stream fills the silo from behind as it is being emptied. Moving the received packet out of the silo into memory does not begin until reception has ceased. In loopback mode, transmit data chaining is not possible. Receive data chaining is allowed regardless of the receive buffer length. (In normal operation, the receive buffers must be 64 bytes long, to allow time for buffer lookahead.) Valid loopback bit settings are as follows:

Loop	INTL	Operation
0	x	Normal on-line operation
1	0	External loopback
1	1	Internal loopback

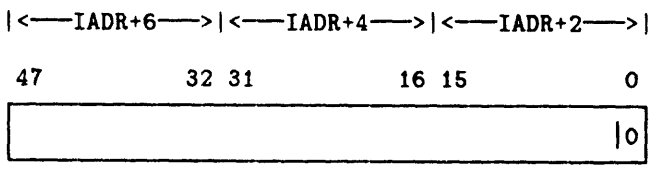
Internal loopback allows the chip to receive its own transmitted packet without disturbing the network. The chip does not receive any packets from the network while it is in internal loopback mode.

Bit	Definition
	External loopback allows the chip to transmit a packet through the transceiver out to the network cable to check the operability of all circuits and connections between the LANCE chip and the network cable. Multicast addressing in external loopback is valid only when DTCR is one (user needs to append the 4 CRC bytes). In external loopback, the chip also receives packets from other nodes.
DTX	Disable transmitter (bit 1). If this bit is set, the chip does not set the TXON bit in NI_CSR0 at the completion of initialization. This prevents the LANCE chip from attempting to access the transmit descriptor ring; hence no transmissions are attempted.
DRX	Disable receiver (bit 0). If this bit is set, the chip does not set the RXON bit in NI_CSR0 at the completion of initialization. This causes the chip to reject all incoming packets and to refrain from attempting to access the receive descriptor ring.

5.10.2 Network Physical Address (NIB_PADR)

The 48-bit physical Ethernet network node address is contained in bytes 2:7 of the initialization block. (This is a network address; it has no relationship to any memory address.) Figure 5-13 shows the network physical address.

Figure 5-13: Network Physical Address (NIB_PADR)

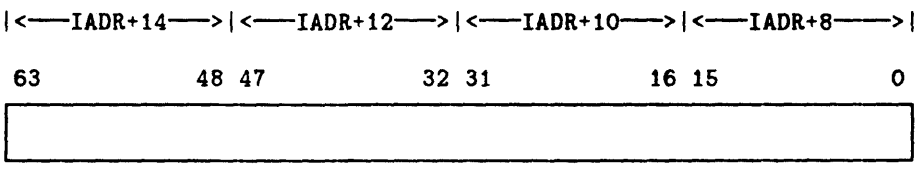


The contents of NIB_PADR identify this station to the network and must be unique within the domain of the network. Its value is normally taken from the network address ROM. The low-order bit (bit 0) of this address must be 0 since it is a physical address.

5.10.3 Multicast Address Filter Mask (NIB_LADRF)

Bytes 8:15 of the initialization block contain the 64-bit multicast address filter mask. The multicast address filter is a partial filter which assists the network controller driver program to selectively receive packets which contain multicast network addresses. Figure 5-14 shows the multicast address filter mask.

Figure 5-14: Multicast Address Filter Mask (NIB_LADRF)



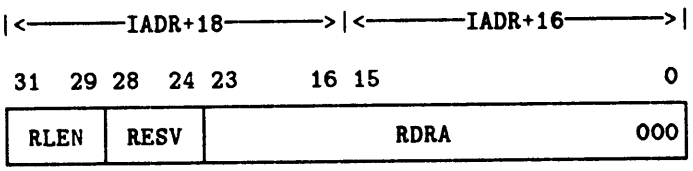
Multicast Ethernet addresses are distinguished from physical network addresses by the presence of a 1 in bit 0 of the 48-bit address field. If an incoming packet contains a physical destination address (bit 0 is 0), then its entire 48 bits are compared with the contents of NIB_PADR and the packet is ignored if they are not equal. If the packet contains a multicast destination address which is all 1's (the broadcast address), it is always accepted and stored regardless of the contents of the multicast address filter mask.

All other multicast addresses are processed through the multicast address filter to determine whether the incoming packet is stored in a receive buffer. This filtering is performed by passing the multicast address field through the CRC generator. The high-order 6 bits of the resulting 32-bit CRC are used to select one of the 64 bits of NIB_LADRF. (These high-order six bits represent in binary the number of the bit in NIB_LADRF, according to the labelling in figure 15-10.) If the bit selected from NIB_LADRF is 1, the packet is stored in a receive buffer; otherwise it is ignored. This mechanism effectively splits the entire domain of 2**47 multicast addresses into 64 parts, and multicast addresses falling into each part are accepted or ignored according to the value of the corresponding bit in NIB_LADRF. The driver program must examine the addresses of the packets accepted by this partial filtering to complete the filtering task.

5.10.4 Receive Descriptor Ring Pointer (NIB_RDRP)

Bytes 16:19 of the initialization block describe the starting address and extent of the receive descriptor ring. Figure 5-15 shows the receive descriptor ring pointer.

Figure 5-15: Receive Descriptor Ring Pointer (NIB_RDRP)



RLEN — Receive ring length (bits 31:29). This field gives the number of entries in the receive descriptor ring, expressed as a power of 2:

RLEN	Entries
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128

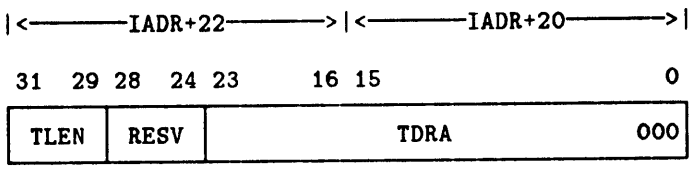
28:24 — Reserved; should be 0's.

RDRA — Receive descriptor ring address (bits 23:0). This is the physical address in system memory of the first element in the ring. Since each 8-byte element must be aligned on a quadword boundary, bits 2:0 of this address must be 0.

5.10.5 Transmit Descriptor Ring Pointer (NIB_TDRP)

Bytes 20:23 of the initialization block describe the starting address and extent of the transmit descriptor ring. Figure 5-16 shows the transmit descriptor ring pointer.

Figure 5-16: Transmit Descriptor Ring Pointer (NIB_TDRP)



TLEN — Transmit ring length (bits 31:29). This field gives the number of entries in the transmit descriptor ring, expressed as a power of 2:

TLEN	Entries
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128

<28:24> — Reserved; should be 0's.

TDRA — Transmit descriptor ring address (bits 23:0). This is the physical address in system memory of the first element in the ring. Since each 8-byte element must be aligned on a quadword boundary, bits 2:0 of this address must be 0.

5.11 Buffer Management

The LANCE chip manages its data buffers by using two rings of buffer descriptors that are stored in memory: the receive descriptor ring and the transmit descriptor ring. Each buffer descriptor points to a data buffer elsewhere in memory, contains the size of that buffer, and contains status information about that buffer's contents.

The starting location in memory of each ring and the number of descriptors in it are given to the LANCE chip through the initialization block during the chip initialization process. Each descriptor is 8 bytes long and must be aligned on a quad-word boundary (the three low-order bits of its address must be 0). The descriptors in a ring are physically contiguous in memory and the number of descriptors must be a power of 2. The LANCE keeps an internal index to its current position in each ring which it increments modulo the number of descriptors in the ring as it advances around each ring.

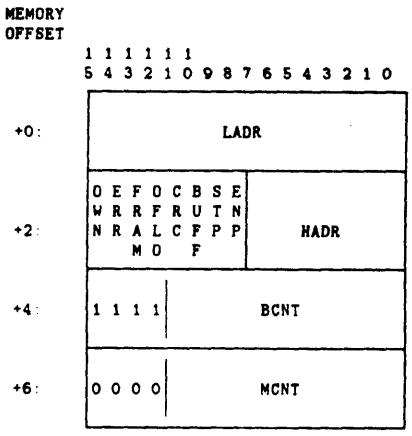
Once started, the LANCE polls each ring to find descriptors for buffers in which to receive incoming packets and from which to transmit outgoing packets, and revises the status information in buffer descriptors as it processes their associated buffers. When polling, the LANCE is limited to looking only one ahead of the descriptor with which it is currently working. The high speed of the data stream requires that each buffer be at least 64 bytes long to allow time to chain buffers for packets which are larger than one buffer. (The first buffer of a packet to be transmitted should be at least 100 bytes to avoid problems in case a late collision is detected.)

Each descriptor in a ring is "owned" either by the LANCE chip or by the host processor; this status is indicated by the OWN bit in each descriptor. Mutual exclusion is accomplished by the rule that each device can only relinquish ownership of a descriptor to the other device, it can never take ownership; and that each device cannot change any field in a descriptor or its associated buffer after it has relinquished ownership. When the host processor sets up the rings of descriptors before starting the Lance, it sets the OWN bits such that the LANCE owns all the descriptors in the receive descriptor ring (to be used by the LANCE to receive packets from the network) and the host owns all the descriptors in the transmit descriptor ring (to be used by the host to set up packets to be transmitted to the network).

5.11.1 Receive Buffer Descriptor

A receive buffer descriptor comprises four words aligned in memory on a quad-word address boundary. See Figure 5-17.

Figure 5-17: Receive Buffer Descriptor



LADR — Low-order buffer address (offset 0, bits 15:0). These are the low-order 16 bits of the 24-bit physical memory address of the start of the buffer associated with this descriptor. Written by the host; unchanged by the LANCE.

HADR — High-order buffer address (offset 2, bits 7:0). These are the high-order 8 bits of the 24-bit physical memory address of the start of the buffer associated with this descriptor. Written by the host; unchanged by the LANCE.

OWN — Owned flag (offset 2, bit 15). This bit indicates whether the descriptor is owned by the host (OWN = 0) or by the LANCE (OWN = 1). The LANCE clears OWN after filling the buffer associated with the descriptor with an incoming packet. The host sets OWN after emptying the buffer. In each case, this must be the last bit changed by the current owner, since changing OWN passes ownership to the other party and the relinquishing party must not thereafter alter anything in the descriptor or its buffer.

ERR — Error summary (offset 2, bit 14). This is the logical OR of the FRAM, OFLO, CRC and BUFF bits in this word. Set by the LANCE and cleared by the host.

FRAM — Framing error (offset 2, bit 13). This bit is set by the LANCE to indicate that the incoming packet stored in the buffer has both a non-integral multiple of eight bits and a CRC error. It is cleared by the host.

OFLO — Overflow error (offset 2, bit 12). This bit is set by the LANCE to indicate that the receiver has lost part or all of an incoming packet because it could not store it in the buffer before the chip's silo overflowed. Cleared by the host.

CRC — Checksum error (offset 2, bit 11). This bit is set by the LANCE to indicate that the received packet has an invalid CRC checksum. Cleared by the host.

BUFF — Buffer error (offset 2, bit 10). This bit is set by the LANCE when it has used all its owned receive descriptors or when it could not get the next descriptor in time while attempting to chain to a new buffer in the midst of a packet. When a buffer error occurs, an overflow error (bit OFLO) also occurs because the LANCE continues to attempt to get the next buffer until its silo overflows. BUFF is cleared by the host.

STP — Start of packet (offset 2, bit 9). This bit is set by the LANCE to indicate that this is the first buffer used for this packet. Cleared by the host.

ENP — End of packet (offset 2, bit 8). This bit is set by the LANCE to indicate that this is the last buffer used for this packet. When both STP and ENP are set in a descriptor, its buffer contains an entire packet; otherwise two or more buffers have been chained together to hold the packet. ENP is cleared by the host.

1111 — Offset 4, bits 15:12 must be set by the host to 1's. Unchanged by the LANCE.

BCNT — Buffer size (offset 4, bits 11:0). This is the number of bytes in the buffer (whose starting address is in HADR and LADR) in 2's complement form. Note that the minimum buffer size is 64 bytes and that the maximum required for a legal packet is 1518 bytes. Written by the host; unchanged by the LANCE.

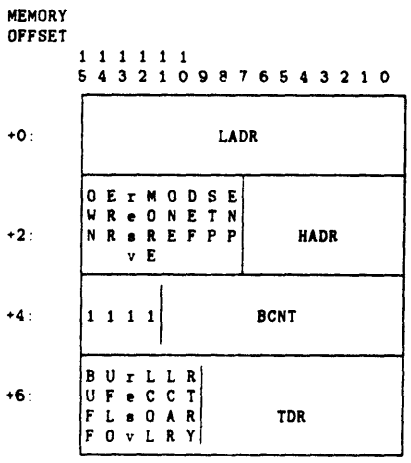
0000 — Offset 6, bits 15:12 are reserved; they should be set to 0's by the host when it constructs the descriptor.

MCNT — Byte count (offset 6, bits 11:0). This is the length in bytes of the received packet for which this is the last or only descriptor. MCNT is valid only in a descriptor in which ENP is set (last buffer) and ERR is clear (no error). Set by the LANCE and cleared by the host.

5.11.2 Transmit Buffer Descriptor

A transmit buffer descriptor comprises four words aligned in memory on a quad-word address boundary. See Figure 5-18.

Figure 5-18: Transmit Buffer Descriptor



LADR — Low-order buffer address (offset 0, bits 15:0). These are the low-order 16 bits of the 24-bit physical memory address of the start of the buffer associated with this descriptor. Written by the host; unchanged by the LANCE.

HADR — High-order buffer address (offset 2, bits 7:0). These are the high-order 8 bits of the 24-bit physical memory address of the start of the buffer associated with this descriptor. Written by the host; unchanged by the LANCE.

OWN — Owned flag (offset 2, bit 15). This bit indicates whether the descriptor is owned by the host (OWN = 0) or by the LANCE (OWN = 1). The host sets OWN after filling the buffer with a packet to be transmitted. The LANCE clears OWN after transmitting the contents of the buffer. In each case, this must be the last bit changed by the current owner, since changing OWN passes ownership to the other party and the relinquishing party must not thereafter alter anything in the descriptor or its buffer.

ERR — Error summary (offset 2, bit 14). This is the logical OR of the LCOL, LCAR, UFLO and RTRY bits in this descriptor. Set by the LANCE and cleared by the host.

Resv — Offset 2, bit 13 is reserved. The LANCE writes a 0 in this bit.

MORE — More retries (offset 2, bit 12). The LANCE sets this bit when more than one retry was required to transmit the packet. Cleared by the host.

ONE — One retry (offset 2, bit 11). The LANCE sets this bit when exactly one retry was required to transmit the packet. Cleared by the host.

DEF — Deferred (offset 2, bit 10). The LANCE sets this bit when it had to defer while trying to transmit the packet. This occurs when the network is busy when the LANCE is ready to transmit. Cleared by the host.

STP — Start of packet (offset 2, bit 9). This bit is set by the host to indicate that this is the first buffer used for this packet. STP is not changed by the LANCE.

ENP — End of packet (offset 2, bit 8). This bit is set by the host to indicate that this is the last buffer used for this packet. When both STP and ENP are set in a descriptor, its buffer contains an entire packet; otherwise two or more buffers have been chained together to hold the packet. ENP is not changed by the LANCE.

1111 — Offset 4, bits 15:12 must be set by the host to 1's. Unchanged by the LANCE.

BCNT — Byte count (offset 4, bits 11:0). This is the number of bytes, in 2's complement form, which the LANCE transmits from this buffer. Note that for any buffer which is not the last of a packet, at least 64 bytes (100 bytes if it is the start of the packet) must be transmitted to allow adequate time for the LANCE to acquire the next buffer. Written by the host; unchanged by the LANCE.

NOTE: *The remaining fields of the descriptor (which make up its entire fourth word) are valid only when the ERR bit in the second word has been set by the LANCE.*

BUFF — Buffer error (offset 6, bit 15). This bit is set by the LANCE during transmission when it does not find the ENP bit set in the current descriptor and it does not own the next descriptor. When BUFF is set, the UFLO bit (below) is also set because the LANCE continues to transmit until its silo becomes empty. BUFF is cleared by the host.

UFLO — Underflow (offset 6, bit 14). This bit is set by the LANCE when it truncates a packet being transmitted because it has drained its silo before it was able to obtain additional data from a buffer in memory. UFLO is cleared by the host.

Resv — Offset 6, bit 13 is reserved. The LANCE writes a 0 in this bit.

LCOL — Late collision (offset 6, bit 12). This bit is set by the LANCE to indicate that a collision has occurred after the slot time of the network channel has elapsed. The LANCE does not retry after a late collision. LCOL is cleared by the host.

LCAR — Loss of carrier (offset 6, bit 11). This bit is set by the LANCE when the carrier-present input to the chip becomes false during a transmission initiated by the LANCE. The LANCE does not retry after such a failure. LCAR is cleared by the host.

RTRY — Retries exhausted (offset 6, bit 10). This bit is set by the LANCE after 16 attempts to transmit a packet have failed due to repeated collisions on the network. (If the DRTY bit of the initialization block MODE word is set, RTRY is set instead after only only one failed transmission attempt.) RTRY is cleared by the host.

TDR — Time domain reflectometer (offset 6, bits 9:0). These bits are the value of an internal counter which is set by the LANCE to count system clocks from the start of a transmission to the occurrence of a collision. This value is useful in determining the approximate distance to a cable fault; it is valid only when the RTRY bit in this word is set.

5.12 LANCE Operation

The LANCE chip operates independently of the host under control of its own internal microprogram. These microcode routines make use of numerous temporary storage cells within the LANCE chip; most of these are not accessible from outside the chip but they are mentioned here when necessary to clarify the operation of the microcode.

Two such (conceptual) internal variables are of central importance: the pointer to the "current" entry in both the receive descriptor ring and in the the transmit descriptor ring. These variables are referred to in this section as TXP and RXP. Each of these designates the descriptor which the LANCE uses for the next operation of that type. If the descriptor designated by one of these pointers is not owned by the LANCE (the OWN bit is 0), then the LANCE can neither perform activity of that type nor advance the pointer. For the transmit ring, the LANCE does nothing until the host sets up a packet in the buffer and sets the OWN bit in the descriptor designated by the Lance's TXP. (The host must keep track of the position of the TXP, since setting up a packet in some other descriptor is not detected by the LANCE). For the receive ring, if the LANCE does not own the descriptor designated by RXP, it cannot receive a packet. In both rings, when the LANCE finishes with a descriptor and relinquishes it to the host by clearing OWN, it then advances the ring pointer (modulo the number of entries in the ring).

When the LANCE begins activity using the current descriptor (the LANCE begins receiving or transmitting a packet), it may look ahead at the next descriptor and attempt to read its first three words in advance so it can chain to the next buffer in mid-packet without losing data. However, it does not actually advance its RXP or TXP until it has cleared the OWN bit in the current descriptor.

5.12.1 Switch Routine

At power-on, the STOP bit is set and the INIT and STRT bits are cleared in NI_CSR0. The LANCE microprogram begins execution in the switch routine, which tests the INIT, STRT, and STOP bits. When the host sets either INIT or STRT, STOP is cleared. If the host writes to NI_CSR1 and NI_CSR2 while STOP is set, that data is stored for use by the initialization routine.

When the microprogram sees STOP cleared, it tests first the INIT bit and then the STRT bit. If INIT is set, it performs the initialization routine. Then if STRT is set, it begins active chip operation by jumping to the look-for-work routine. Control returns to the switch routine whenever the host again sets the STOP bit (which also clears the INIT and STRT bits). Note that the ring pointers RXP and TXP are not altered by the setting of either STOP or START; they are reset to the start of their rings only when INIT is set.

5.12.2 Initialization Routine

The initialization routine is called from the switch routine when the latter finds the INIT bit set. It reads the initialization block from the memory addressed by NI_CSR1 and NI_CSR2 and stores its data within the LANCE chip. This routine also sets the ring pointers RXP and TXP to the start of their rings (that is, at the lowest memory address in the ring).

5.12.3 Look-For-Work Routine

The look-for-work routine is executed while the LANCE is active and looking for work. It is entered from the switch routine when the STRT bit is set, and is returned to from the receive and transmit routines after they have received or transmitted a packet.

This routine begins by testing whether the receiver is enabled (bit RXON of NI_CSR0 is set). If so, it tries to have a receive buffer available for immediate use when a packet addressed to this system arrives. The routine tests its internal registers to see whether it has already found a receive descriptor owned by the LANCE and, if not, calls the receive poll routine to attempt to get a receive buffer.

Next the routine tests whether the transmitter is enabled (bit TXON of NI_CSR0 is set). If so, it calls the transmit poll routine to see whether there is a packet to be transmitted. If a packet is available, the transmit poll routine transmits it.

If there is no transmission and the TDMD bit of NI_CSR0 is not set, the microprogram delays 1.6 milliseconds and then goes to check the receive descriptor status again. If a packet was transmitted or the host has set TDMD, the delay is omitted so that multiple packets are transmitted as quickly as possible.

If at any point in this routine the receiver detects an incoming packet whose destination address matches the station's physical address, or matches the broadcast address, or passes the multicast address filter (or if the PROM bit of NIB_MODE is set), the receive routine is called.

5.12.4 Receive Poll Routine

The receive poll routine is called whenever the receiver is enabled and the LANCE needs a free buffer from the receive descriptor ring. The routine reads the second word of the descriptor designated by RXP and, if the OWN bit the second word is set, the routine reads the first and third words also.

5.12.5 Receive Routine

The receive routine is called when the receiver is enabled and an incoming packet's destination address field matches one of the criteria described in Section 5.12.3. The routine has three sections: initialization, lookahead, and descriptor update.

In initialization, the routine checks whether a receive ring descriptor has already been acquired by the receive poll routine. If not, it makes one attempt to get the descriptor designated by RXP (if OWN is not set in the descriptor, MISS and ERR are set in NI_CSR0 and the packet is lost). The buffer thus acquired is used by the receive DMA routine to empty the silo.

In lookahead, the routine reads the second word of the next descriptor in the receive ring and, if the OWN bit is set, reads the rest of the descriptor and holds it in readiness for possible data chaining.

The descriptor update section is performed when either the current buffer is filled or the packet ends. If the packet ends but its total length is less than 64 bytes, it is an erroneous *runt packet* and is ignored: no status is posted in the descriptor, RXP is not moved, and the buffer is reused for the next incoming packet (this is why a receive buffer must be at least 64 bytes long; otherwise the runt might be detected after advancing RXP).

If the packet ends (with or without error), the routine writes the packet length into MCNT, sets ENP and other appropriate status bits and clears OWN in the current descriptor, and sets RINT in NI_CSR0 to signal the host that a complete packet has been received. Then it advances RXP and returns to the look-for-work routine.

If the buffer is full and the packet has not ended, chaining is required. The routine releases the current buffer by writing status bits into its descriptor (clearing OWN and ENP, in particular), makes current the next descriptor data acquired in the lookahead section, advances RXP, and goes to the lookahead section to prepare for possible additional chaining. Note that RINT is not set in NI_CSR0, although the host would find OWN cleared if it looked at the descriptor, and it could begin work on that section of the packet, since the mutual exclusion rule prevents the LANCE from going back and altering it.

5.12.6 Receive DMA Routine

The receive DMA routine is invoked asynchronously by the chip hardware during execution of the receive routine whenever the silo contains 16 or more bytes of incoming data or when the packet ends and the silo is not empty. It executes DMA cycles to drain data from the silo into the buffer designated by the current descriptor.

5.12.7 Transmit Poll Routine

The transmit poll routine is called by the look-for-work routine to see whether a packet is ready for transmission. It reads the second word of the descriptor designated by TXP and tests the OWN bit. If OWN is 0, the LANCE does not own the buffer and this routine returns to its caller. If OWN is set, the routine tests the STP bit, which should be set to indicate the start of a packet. If STP is clear, this is an invalid packet; the LANCE sets its OWN bit to return it to the host, sets TINT in NI_CSR0 to notify the host, and advances TXP to the next transmit descriptor. If both OWN and STP are set, this is the beginning of a packet, so the transmit poll routine reads the rest of the descriptor and then calls the transmit routine to transmit the packet. During this time the chip is still watching for incoming packets from the network and it aborts the transmit operation if one arrives.

5.12.8 Transmit Routine

The transmit routine is called from the transmit poll routine when the latter finds the start of a packet to be transmitted. The transmit routine has three sections: initialization, lookahead, and descriptor update.

In initialization, the routine sets the chip's internal buffer address and byte count from the transmit descriptor, enables the transmit DMA engine, and starts transmission of the packet preamble. It then waits until the transmitter is actually sending the bit stream (including possible backoff-and-retry actions in case of collisions).

In lookahead, the transmit routine tests the current descriptor to see whether it is the last in the packet (the ENP bit is set). If so, no additional buffer is required so the routine waits until all the bytes from the current packet have been transmitted. If not, the routine attempts to get the next descriptor and hold it in readiness for data chaining, and then waits until all the bytes from the current buffer have been transmitted.

Descriptor update is entered when all the bytes from a buffer have been transmitted or an error has occurred. If there is no error and the buffer was not the last of the packet, the pre-fetched descriptor for the next buffer is made current for use by the transmit DMA routine. The routine writes the appropriate status bits and clears the OWN bits in the current descriptor and advances TXP. If this was the last buffer in the packet, the routine sets the TINT bit in NI_CSR0 to notify the host and returns to the look-for-work routine; otherwise it goes back to the lookahead section in this routine.

5.12.9 Transmit DMA Routine

The transmit DMA routine is invoked asynchronously by the chip hardware during execution of the transmit routine whenever the silo has 16 or more empty bytes. It executes DMA cycles to fill the silo with data from the buffer designated by the current descriptor.

5.12.10 Collision Detect Routine

This routine is invoked asynchronously by the chip hardware during execution of the transmit routine when a collision is detected on the network. It ensures that the *jam* sequence is transmitted, then backs up the chip's internal buffer address and byte count registers, waits for a pseudo-random backoff time, and then attempts the transmission again. If 15 retransmission attempts fail (a total of 16 attempts), it sends the microcode to the descriptor update routine to report an error in the current transmit descriptor (bits RTRY and ERR are set).

5.13 LANCE Programming Notes

1. The interrupt signal is the OR of the interrupt-causing conditions. If another such condition occurs while the interrupt signal is already asserted, there is not another active transition of the interrupt signal and the interrupt request bit in INT_REQ is not set again. An interrupt service routine should use logic similar to the following to avoid losing interrupts:

   - Read NI_CSR0 and save the results in a register (for example, R0).
   - Clear the interrupt enable bit INEA in the saved data in R0.
   - Write NI_CSR0 with the saved data in R0. This makes the interrupt signal false because INEA is clear and clears all the write-one-to-reset bits such as RINT, TINT and the error bits; this process does not alter the STRT, INIT or STOP bits nor any interrupt-cause bits which come true after NI_CSR0 was read.
   - Write NI_CSR0 with only INEA to enable interrupts again.
   - Service all the interrupt and error conditions indicated by the flags in the data in R0.
   - Exit from the interrupt service routine.
   - Be sure to access NI_CSR0 only with instructions which do a single access, such as MOVE. Instructions such as BIS which do a read-modify-write operation can have unintended side effects.

2. An interrupt is signalled to the host only when the last buffer of a multi-buffer (chained) packet is received or transmitted. However, the OWN bit in each descriptor is cleared as soon as the LANCE has finished with that portion of the packet, and the mutual exclusion rule makes it safe for the host to process such a descriptor and its buffer.

3. When a transmitter underflow occurs (UFLO is set in a transmit descriptor because the silo is not filled fast enough), the LANCE turns off its transmitter and the LANCE must be restarted to turn the transmitter back on again. This can be done by setting STOP in NI_CSR0 and then setting STRT in NI_CSR0 (DTX is still clear in the chip's internal copy of NIB_MODE). It is not necessary to set INIT to reread the initialization block.

   Note that setting STOP immediately terminates any reception which is in progress. If the status of a receive descriptor has been updated and its OWN bit is now clear, then the contents of its buffer are valid. If the incoming packet was chained into more than one buffer, however, the

packet is only valid if its last buffer has been completed (the one with the ENP bit set).

4. The network controller hardware requires up to five seconds after power on to become stable. Self-test routines must delay at least five seconds before attempting to use the controller for either internal or external testing.

5. The LCAR flag (loss of carrier) may be set in the transmit descriptor when a packet is sent in internal loopback mode. When the LANCE is operating in internal loopback mode and a transmission is attempted with a non-matching address, the LANCE correctly rejects that packet. If the next operation is an internal loopback transmission, and the LANCE has not been reset, the packet is not sent and LCAR is set in the transmit descriptor for that packet. The receive descriptor is still owned by the LANCE. To avoid this problem, the LANCE should be reinitialized after each internal loopback packet.

6. The one flag is occasionally set in a transmit descriptor after a late collision. The LANCE does not attempt a retransmission even though one may be set. The host should disregard one if the LCOL flag is also set.

7. The chip's internal copy of NI_CSR1 may become invalid when the chip is stopped. The NI_CSR1 and NI_CSR2 registers should always be loaded prior to setting INIT to initialize the LANCE chip.

8. Attempting an external loopback test on a busy network can cause a silo pointer misalignment if a transmit abort occurs while the chip was preparing to transmit the loopback packet. The resulting retransmission may cause the transmitter enable circuit to hang, and the resulting illegal length transmission must be terminated by the jabber timer in the transceiver. It is unlikely that there may be a corrupted receive buffer because the reception that caused the transmit abort usually does not pass address recognition.

   Since external loopback is a controlled situation, it is possible to implement a software procedure to detect a silo pointer misalignment problem and prevent continuous transmissions. Because the test is being done in loopback, the exact length and contents of the receive packet are known; thus the software can determine whether the data in the receive buffer has been corrupted.

9. When the chip is in internal loopback mode and a CRC error is forced, a framing error is indicated along with the CRC error. In external loopback, when a CRC error is forced only that error is indicated; a framing error is indicated only if the LANCE actually receives extra bits.

10. When transmit data chaining, a BUFF error is set in the current transmit descriptor if a late collision or retry error occurred while the LANCE was still transmitting data from the previous buffer. The BUFF error in this case is an invalid error indication and should be ignored. BUFF is valid only when UFLO is also set.

11. When the host program sets up a packet for transmission in chained buffers, it should set the OWN bits in all the transmit buffers except the first one (that is, the one containing the STP bit), and then as its last act, the host program should set the OWN bit in the first descriptor. Once that bit is set, the LANCE starts packet transmission and may encounter an underflow error if the subsequent descriptors for the packet are not available.

12. INIT and STRT should not be set in NI CSR0 at the same time. After stopping the chip, first set INIT and wait for IDON, then set STRT. If both are set at once, corrupt transmit or receive packets can be generated if RENA becomes true during the initialization process.

5.14 Power Requirements

The DESVA requires 5 volts with a tolerance of plus or minus five percent. The typical current drawn is 1.0 amps.

Chapter 6
Resistor Load Module

The system box must use a resistor load module when less than two drives are installed. The resistor load module regulates the power supply in the expansion boxes when only one drive is installed in each box. The power supply needs a minimum amount of current drawn for it to regulate properly. The single disk in the hard disk expansion box and the tape drive with the controller board in the tape expansion box do not draw enough current for the power supply to regulate. The resistor load module is installed in these boxes to draw a sufficient amount of current to allow the power supply to regulate properly. Figure 6-1 shows the resistor load module and Figure 6-2 shows the circuit diagram of the resistor load module.

The +5 Vdc portion of the load module draws 3 Amps and the +12 Vdc portion draws 1 Amp. The module measures 7 inches (177.8mm) by 4 inches (101.6mm).

Figure 6-1: Resistor Load Module

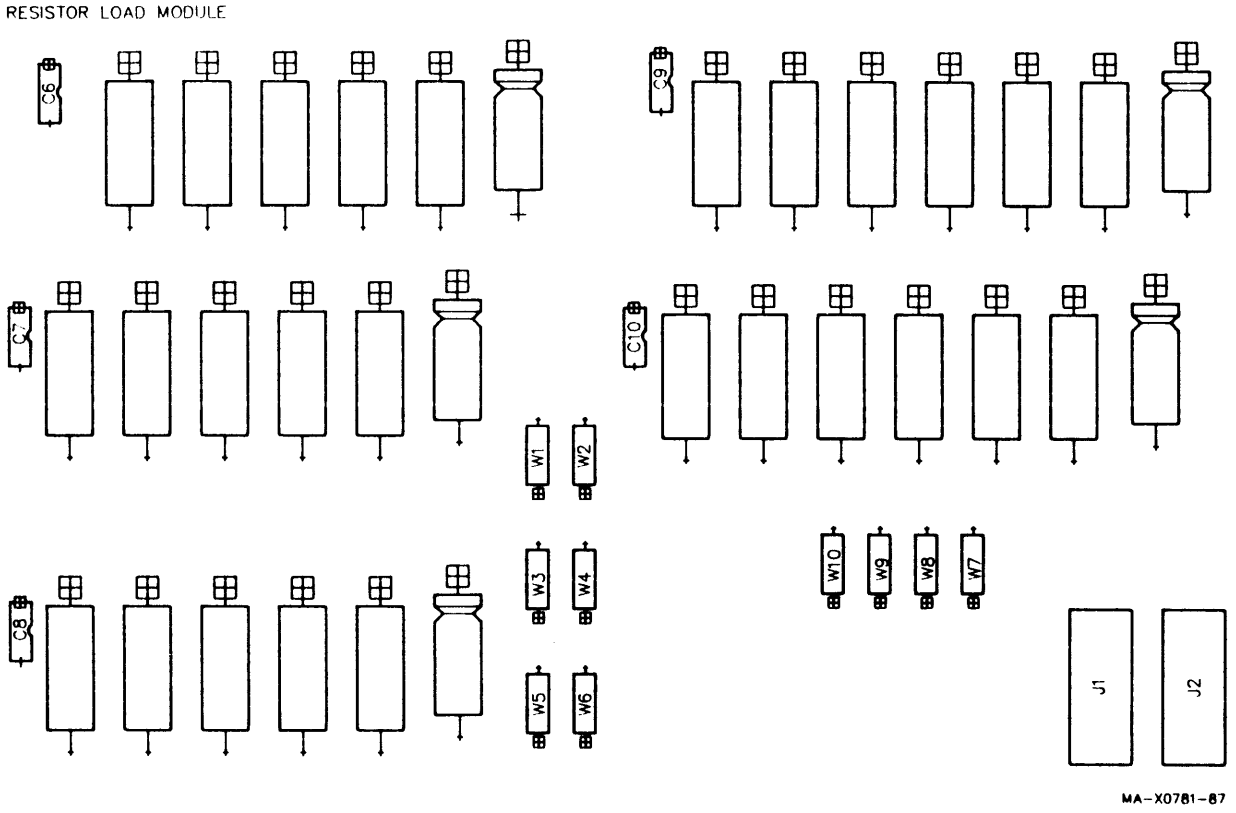
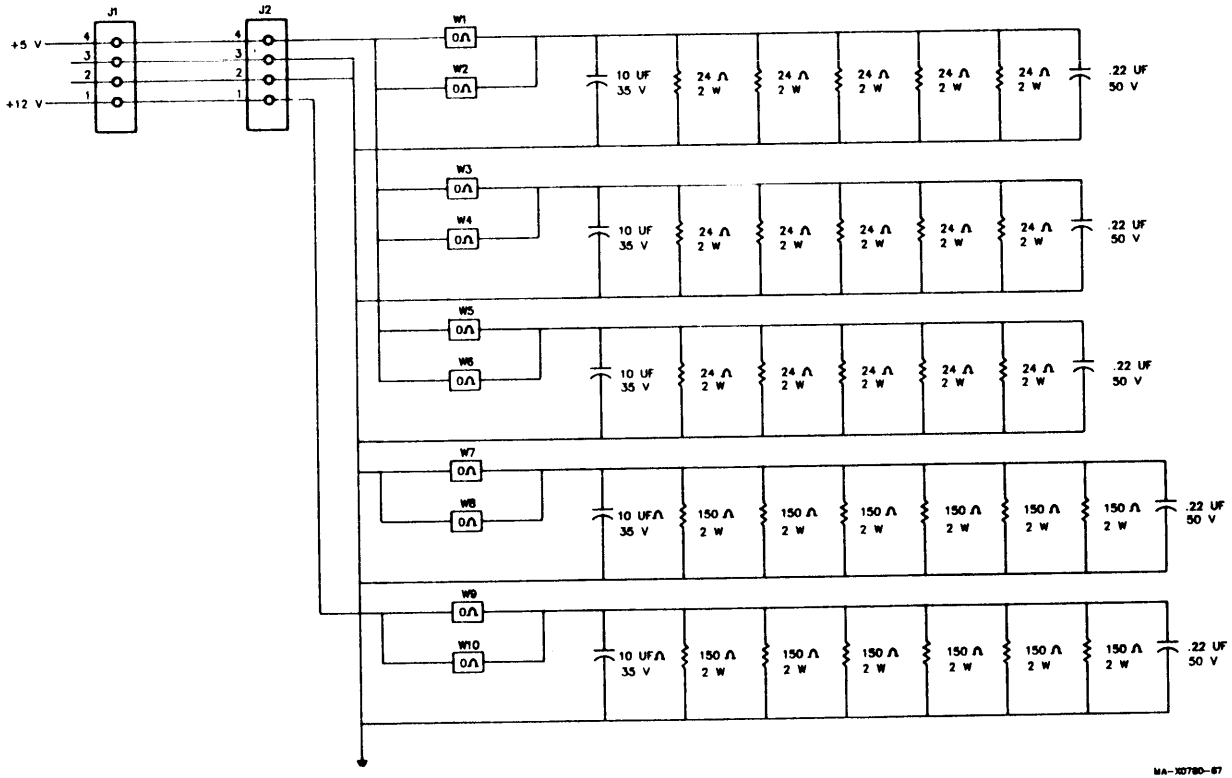


Figure 6–2: Resistor Load Module Circuit Diagram



Chapter 7
Power Supply

7.1 Introduction

The VS410 system box and each VS40B storage expansion box are powered by an H7848 power supply. Model H7848-AA is for nominal 115 V input and model H7848-AB is for nominal 230 V input. The power supply assembly includes an ac power connector, an ac power switch, and a variable-speed cooling fan.

7.2 AC Input

Single-phase ac power is supplied through a 3-pin IEC 320 C14 connector for a BCC02-xx power cord, where the variable is appropriate to national usage. Table 7-1 lists the input power specifications.

Table 7-1: AC Input Specifics

Model	Minimum	Nominal	Maximum
Input voltage (single phase)			
H7848-BA	88	100 - 120	132 Vac rms
H7848-BB	176	220 - 240	264 Vac rms
Frequency			
-BA and -BB	47	50 - 60	63 Hertz
Miscellaneous			
Power input: 160 watts maximum.			
Power factor: 0.6 minimum.			

Table 7–1 (Cont.): AC Input Specifics

Model	Minimum	Nominal	Maximum
Inrush current: 32 amps maximum for one-half AC cycle.			
Steady state RMS current:			
	2.4 amps in 100-120 volt range		
	1.3 amps in 220-240 volt range		

7.3 DC Output

Table 7–2 lists the output power specifications.

Table 7–2: DC Output Specifications

Nominal Voltage	Min. Voltage	Max. Voltage	Max. Noise Less Than 10 MHz (mVolts)	Max. Noise Greater Than 10 MHz (percentage)	Min. Amps	Max. Amps
+5.10	+4.85	+5.35	50.0	3.0	3.00	10.00[1]
+12.10	+11.50	+12.70	70.0	2.0	0.50	4.00[1]
-12.00	-11.40	-12.60	120.0	2.0	0.00	0.25
-9.00[2]	-8.55	-9.45	50.0	2.0	0.00	0.20

Maximum output power: 104 watts

[1]If the +12.1 Vdc is limited to 3.0 Amps maximum, the +5.1 Vdc can supply up to 12.0 Amps. If the +12.1 Vdc is limited to 2.0 Amps maximum, the +5.1 Vdc can supply up to 13.0 Amps.

[2]Isolated supply

7.4 Battery for Time-of-Year Clock

When the system is powered off, the time of year clock and its associated 50 bytes of RAM storage are powered by a three-cell nickel cadmium (NiCad) battery pack (part number 12-19245-00), which is rated to supply 3.6 V and has a capacity of 180 milliampere hours.

7.5 Cooling

The airflow intake passes through a grill in the front panel of the enclosure which extends the full width of the unit, above the disk drive access door and the ac power switch. The airflow exhaust passes through a grill in the rear enclosure panel (at the right side when viewed from the rear). There are no air vents in the top or bottom, or in either side panel of the enclosure.

Chapter 8
Drives

8.1 Introduction

This chapter provides an overview of the drives that are currently available for use with the VAXstation 2000 and MicroVAX 2000 systems. Refer to the technical description manual on each drive for a detailed description. Table 8–1 lists the drives covered in this chapter and their technical description manual order number.

Table 8–1: Drives

Drive	Manual Order Number
RX33 half-height diskette drive	EK-RX33T-TM
RD32 half-height hard disk drive	EK-RD32A-TD
RD53 full-height hard disk drive	EK-RD53A-TD
TK50 tape drive	EK-TZK50-TM

8.2 RX33 Half-Height Diskette Drive

The RX33 is a 5.25 inch, double-sided, half-height diskette drive. It has two operating speeds: for normal and for high-density diskettes (up to 96 tracks per inch). The RX33 provides full read/write compatibility with an RX50 single-sided drive. Figure 8–1 shows the top and front view of the RX33 diskette drive. This drive can only be installed in the system box. Figure 8–2 shows the RX33 diskette.

Figure 8–1: RX33 Diskette Drive

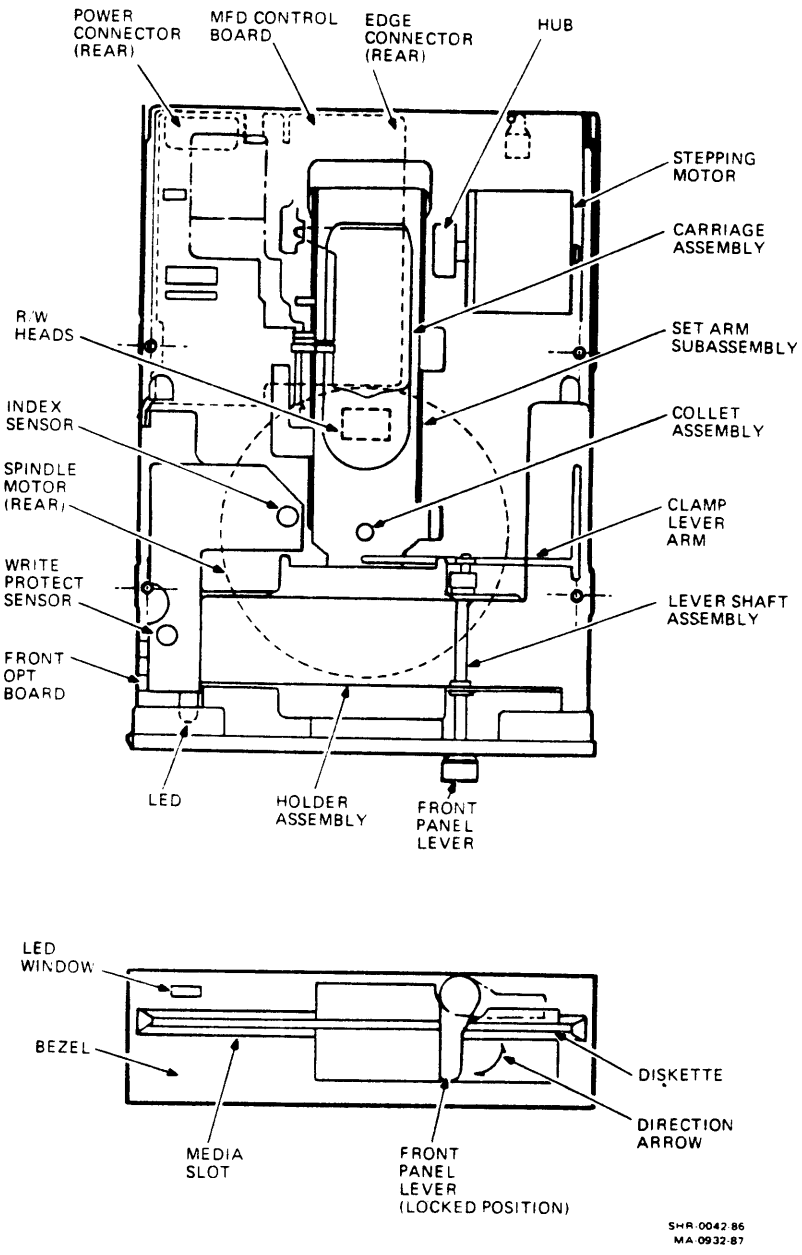
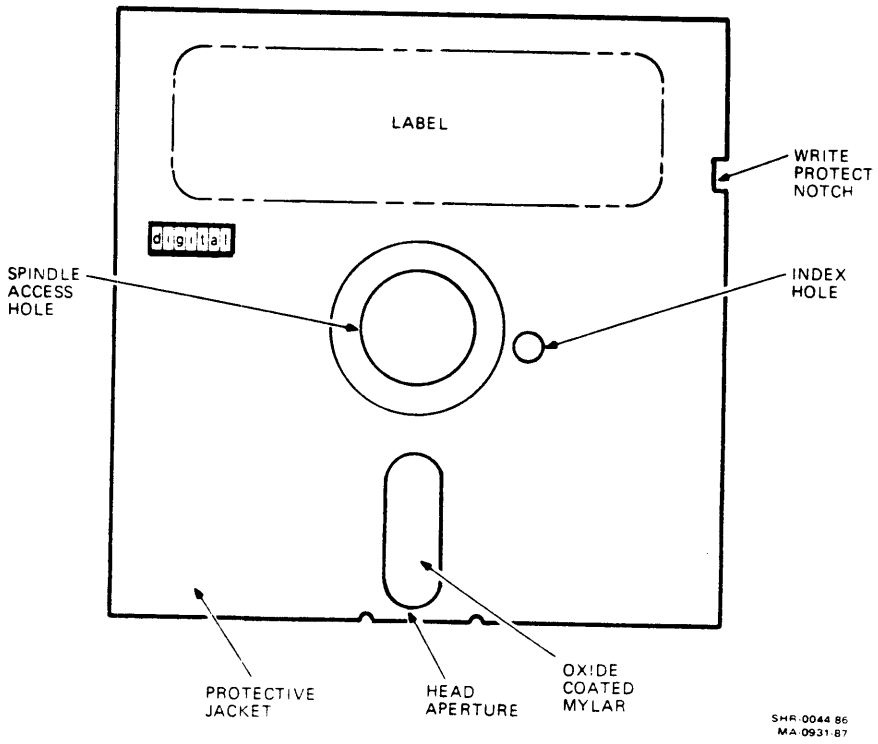


Figure 8-2: RX33 Diskette



8.2.1 RX33 Media

The RX33 uses 130 mm (5.25 inch) soft-sectored diskettes. These diskettes can be single-sided or double-sided. They can also be high or normal density. The type of operating mode selected (high or normal) depends on the diskette inserted in the drive.

Operating Mode	Diskette Required
Normal density	Single-sided, normal-density diskette (RX50-type), 96 tracks per inch
High Density	Double-sided, high-density diskette (RX33-type)

The two operating modes use different data transfer rates.

Operating Mode	Data Transfer Rate
Normal Density	250 kilobits per second
High Density	500 kilobits per second

8.2.2 RX33 Jumper Configuration

The following jumpers must be installed for normal operation. Figure 8-3 shows the RX33 with the proper jumpers installed.

Jumpers provide the following functions.

Jumper	Description
DS0	Selects drive 0
HG and I	Allows the disk controller control the operating mode (normal or high density)
FG	Provides frame grounding
DC	Diskette change mode
Bus Terminator	Must be installed for proper communication

8.2.3 Inserting/Removing a Diskette

The RX33 has a single diskette slot in its front panel. You can insert a diskette as follows.

1. Make sure the diskette's label is facing up and the write-protect notch is on the left as shown in Figure 8-4.
2. Push the diskette into the slot, until the diskette snaps into position.
3. Lock the front panel lever by turning the lever 90 degrees to the left (counterclockwise).

   CAUTION: *Do not force the lever. You can only turn the lever when a diskette is fully inserted in the drive.*

To remove a diskette, simply turn the front panel lever 90 degrees to the right (clockwise). The diskette springs out for easy removal.

CAUTION: *Do not open the lever if the LED indicator on the front panel is on. Hard write errors may result.*

Figure 8-3: RX33 Jumper Settings

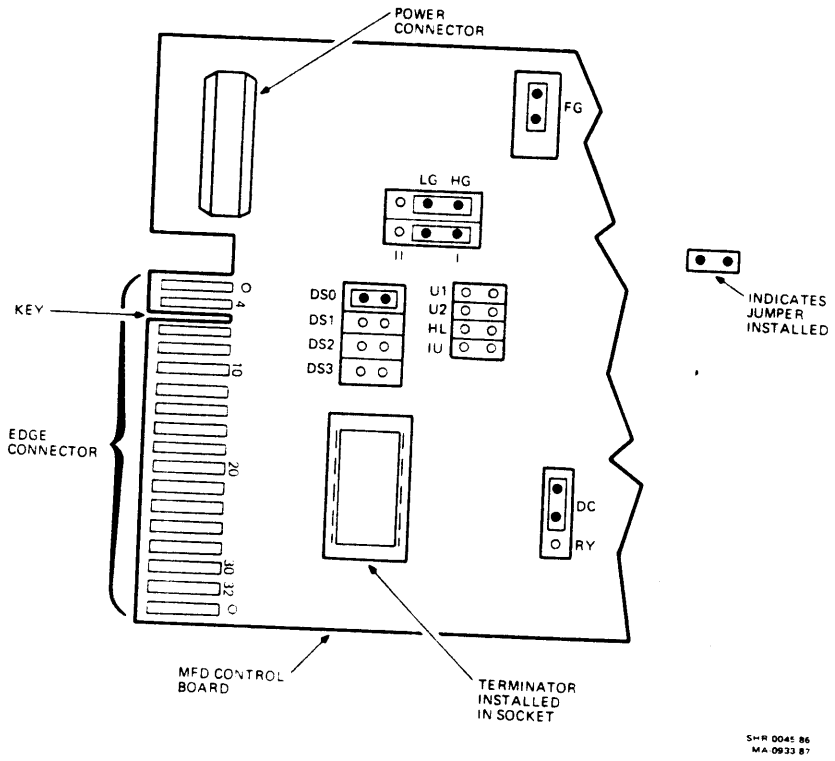
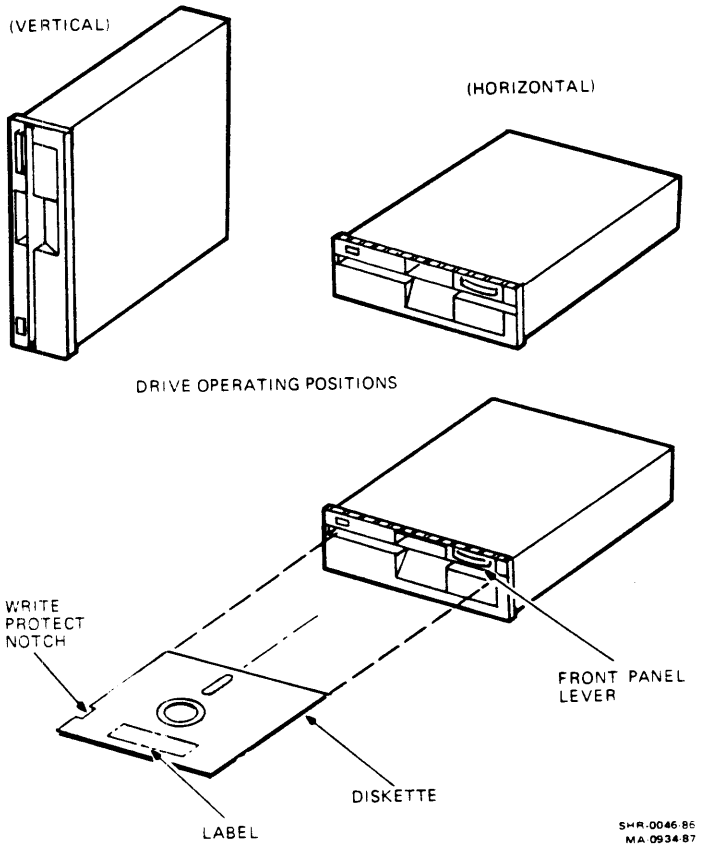


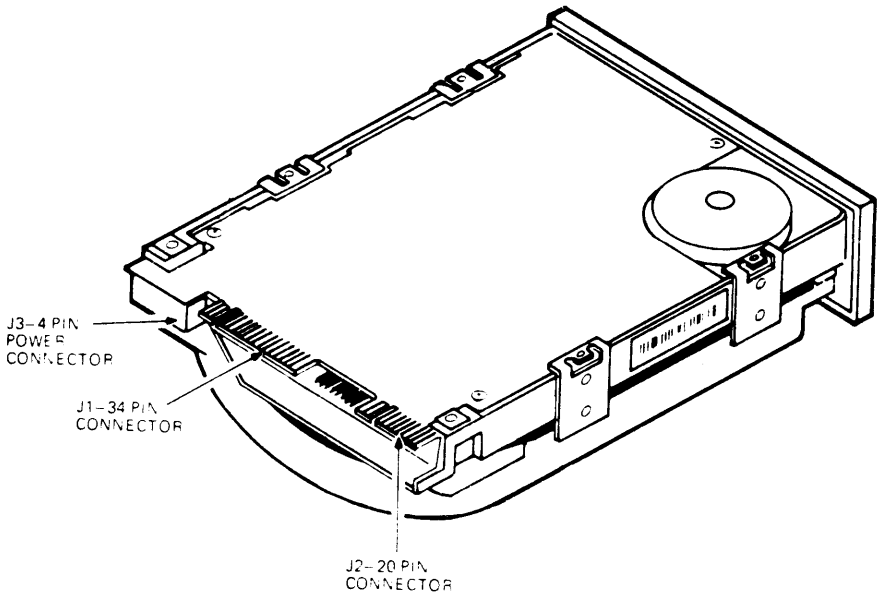
Figure 8-4: Inserting a Diskette



8.3 RD32 Half-Height Hard Disk Drive

The RD32 is a half-height hard disk drive. This drive contains 42 megabytes of memory when formatted. It is usually installed in the system box along with the RX33 floppy diskette drive but can also be installed in the hard disk expansion box. Figure 8–5 shows the connectors on the back of the RD32 disk drive.

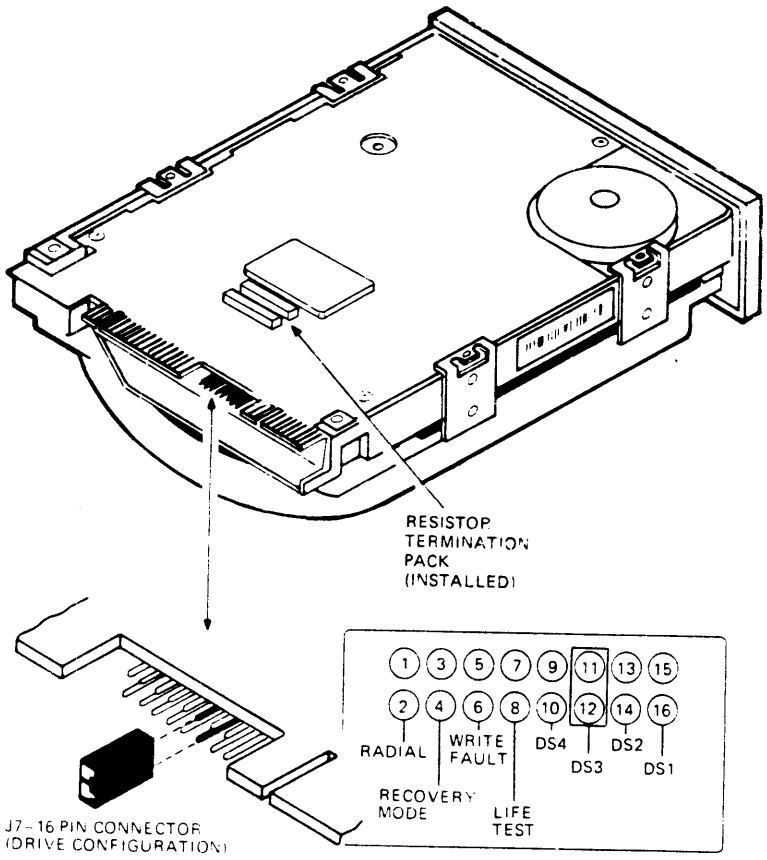
Figure 8–5: RD32 Power and Data Connectors



8.3.1 RD32 Jumper Configuration

There is only one configuration setting for the jumpers on the RD32 device electronics board when it is used in the VAXstation 2000 or the MicroVAX 2000 systems. Also, the same jumper setting is used whether the drive is installed in the system box or in the expansion box. Figure 8–6 shows the location and configuration of the jumpers on the RD32 device electronics board.

Figure 8–6: RD32 Jumper Configuration



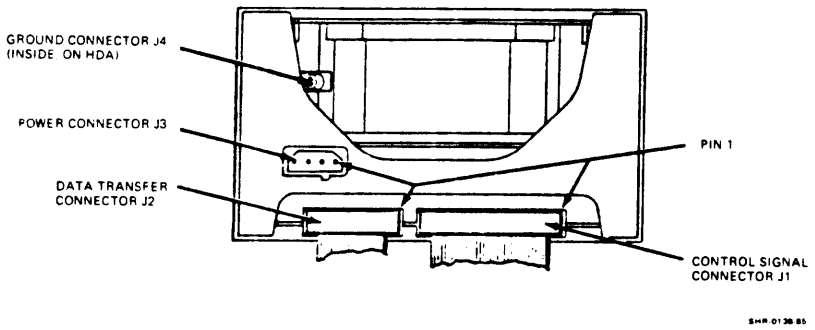
SHOWN WITH DRIVE CONFIGURED AS DS–1

SHR 0418 85
MA-0131-87

8.4 RD53 Full-Height Hard Disk Drive

The RD53 is a full-height hard disk drive. This drive contains 71 megabytes of memory when formatted. This drive can be installed in the system box or in the hard disk expansion box. Installing the RD53 in the system box prevents the installation of any other drive within the system box. Figure 8–7 shows the connectors on the back of the RD53 disk drive.

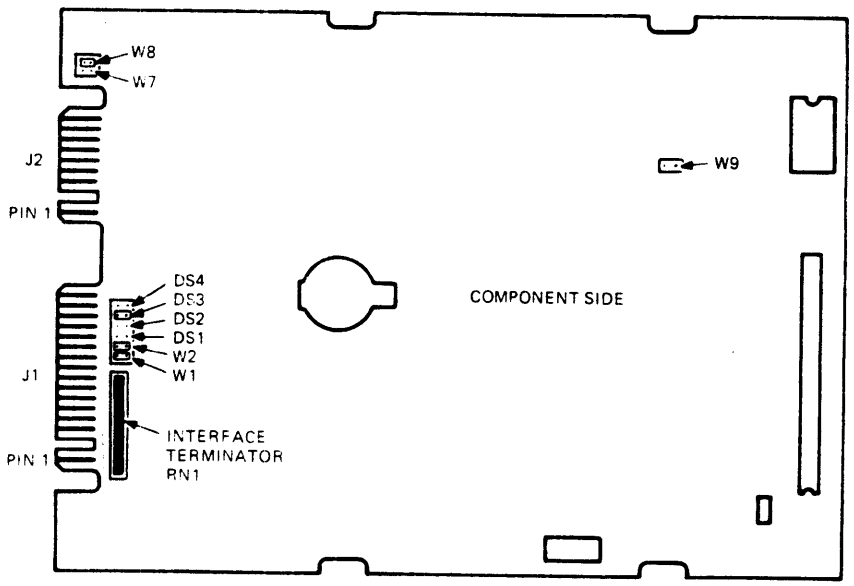
Figure 8–7: RD53 Power and Data Connectors



8.4.1 RD53 Jumper Configuration

There is only one configuration setting for the jumpers on the RD53 device electronics board when it is used in the VAXstation 2000 or the MicroVAX 2000 systems. Also, the same jumper setting is used whether the drive is installed in the system box or in the expansion box. Figure 8-8 shows the location and configuration of the jumpers on the RD53 device electronics board.

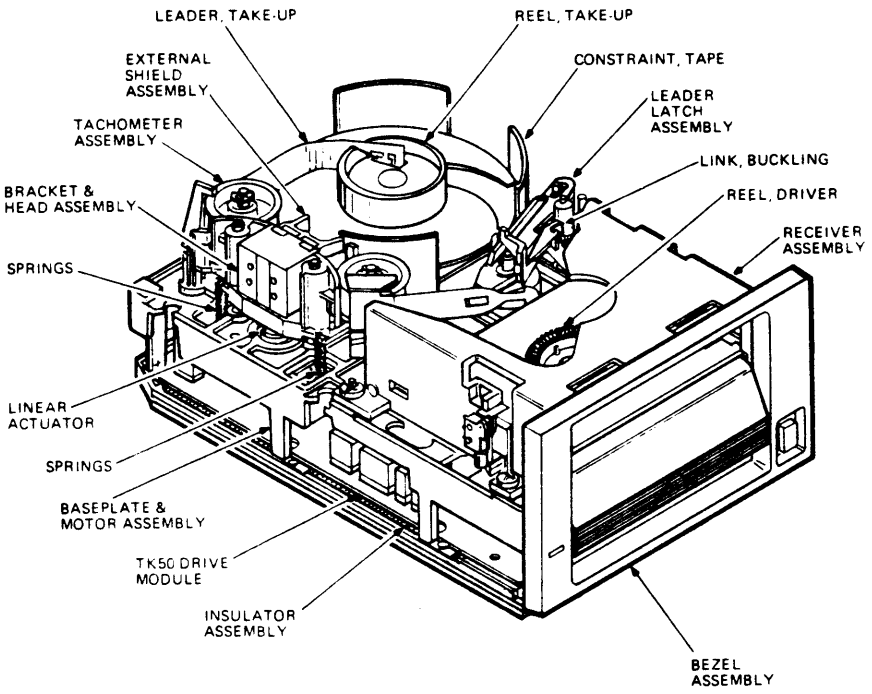
Figure 8-8: RD53 Jumper Configuration



8.5 TK50 Tape Drive

The TK50 tape drive is a mass storage device. This drive can only be installed in the tape expansion box. The system is not capable of supporting the TK50 drive within the system box. The drive uses removable 94.5 megabyte tape cartridges to provide backup storage and software distribution for the VAXstation 2000 and the MicroVAX 2000 systems. The storage medium is a tape cartridge containing a magnetic tape that is 0.5 inch wide and 600 feet long. The tape cartridge is about 4 by 4 inches square, and is labeled CompacTape. Figure 8-9 shows a cutaway view of the TK50 tape drive.

Figure 8-9: Cutaway View of the TK50 Tape Drive



8.5.1 Using the TK50

The load/Unload push button switch controls the TK50 tape drive. A green indicator light shows activity in the drive and a red LED in the load/unload switch shows the operating status of the drive. Figure 8-10 shows the front of the TK50. The rear panel has the logic and power connectors as shown in Figure 8-11.

Figure 8-10: TK50 Front View

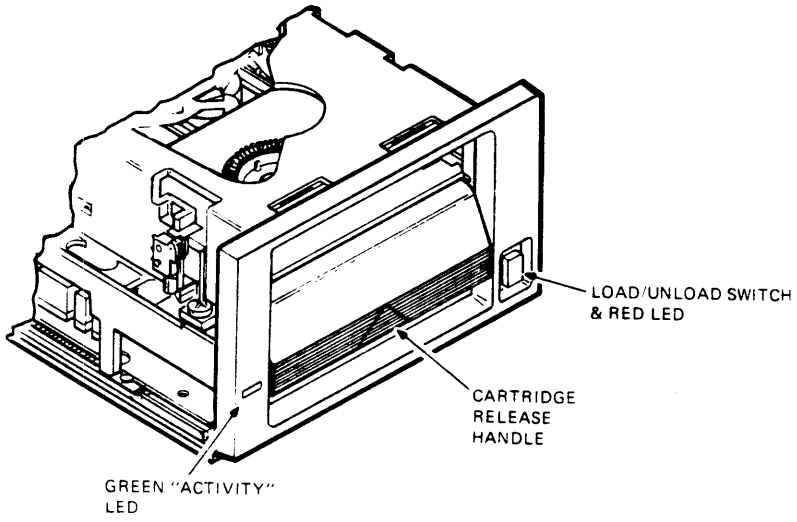
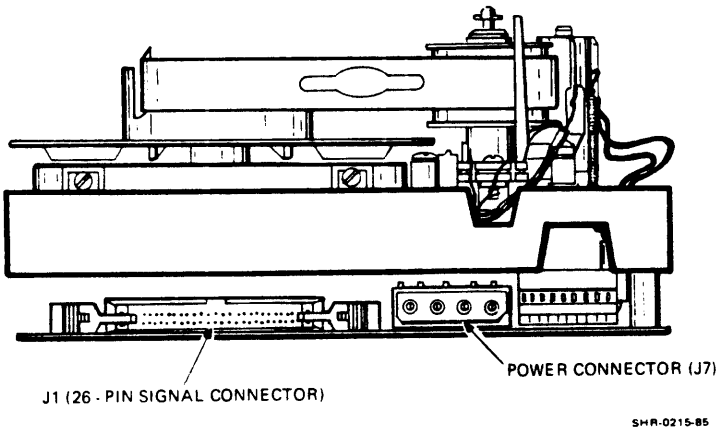


Figure 8–11: TK50 Rear View



8.5.1.1 Loading/Unloading a Tape Cartridge

To load a tape, do the following.

1. Make sure the Load/Unload switch is in the *out* position.
2. Power-up the tape expansion box. The TK50 performs its power-up self-test (about five seconds). When no cartridge is in the drive, the red light in the Load/Unload switch turns on during power-up. On successful completion of the self-test, the red light turns off and the green LED turns on. The drive is now ready to load.
3. Lift the handle.
4. Insert the cartridge all the way into the drive. When the cartridge is most of the way in, the red light turns on and the green LED turns off.
5. Lower the handle. The red light turns off and the green LED turns on.
6. Press the Load/Unload switch to the *in* position. The red light turns on and the green LED turns off.
7. The tape is now being loaded to the beginning of tape. When the tape is successfully loaded, the green LED turns on. The green LED blinks when the drive is seeking the correct position of the tape and also when the drive is reading or writing.

To unload a tape, do the following.

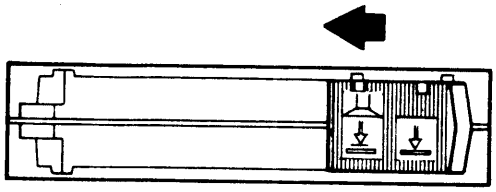
1. Press the Load/Unload switch.
2. When the tape is completely rewound and unloaded, the red light turns off and the green LED turns on. Both of these indicators blink as the tape rewinds.
3. List the handle.
4. Remove the cartridge.

NOTE: *Always remove a cartridge from the drive before powering down the tape expansion box. Otherwise, you cannot remove the cartridge once power is removed from the drive.*

8.5.2 Write Protecting a TK50 Tape Cartridge

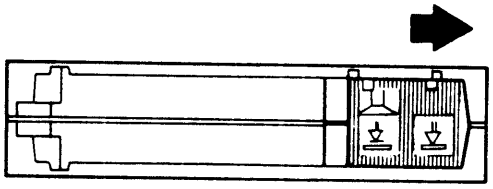
Slide the Write Protect switch to the left to write protect the tape as shown in Figure 8-12. Slide the Write Protect switch to the right to disable write protect as shown in Figure 8-13.

Figure 8-12: Write Protecting a Tape Cartridge



SHR-0310-84

Figure 8-13: Disabling Write Protect on a Tape Cartridge



SHR-0311-84

Chapter 9

DEC423 Converter (MicroVAX 2000)

9.1 Introduction

The DEC423 converter changes the three RS232 ports on the video and printer connectors to three DEC423 modified modular jacks (MMJ). DEC423 is a superset of RS423. This communication strategy is supported through the DECconnect terminal interconnect system (DTIS) which permits easy installation of terminals and printers using the MMJ connectors and cabling similar to that used for telephone installation. Terminals that currently use the RS232 protocol which do not have MMJ connectors are connected to DTIS using active converters (H3105) or passive adapters (H8571-A, 25-pin, and H8571-B, 9-pin). The DEC423 converter assembly measures 3 x 3.3 x 1.23 inches. It mounts directly to the video and printer connectors on the back of the system box and provides the following features.

- Conversion from D-sub connectors and RS232, to DEC423 and MMJs for three of the serial lines on the back of the system box.
- Electrostatic discharge (ESD) and Electrical overstress (EOS) protection.
- FCC Part 15 qualification for use with unshielded DTIS cable.
- Power is received from the system box.
- Modem control is not supported.

9.2 Physical Description

This section describes the physical characteristics of the DEC423 converter.

9.2.1 Converter Enclosure

The converter enclosure consists of a two-piece plastic housing with a PC card inside. Two D-sub connectors, three MMJ connectors, and all the circuitry are contained on this board. The design of the plastic housing is such that the interior can be metalized for shielding in special applications if necessary, with positive connection to the PC card ground plane. The size of the enclosure measures 3 x 3.3 x 1.23 inches.

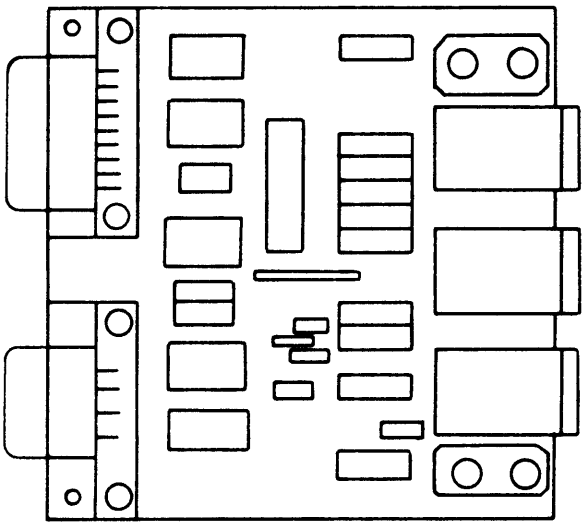
9.2.2 Mounting

The converter is secured directly over the RS232 D-sub connectors on the back of the system box. The D-sub connectors are keyed and are different sizes, so it is impossible for the converter to be attached wrong. Unshielded DTIS MMJ cables (up to 3) are then plugged into the converter for attachment to user terminals and equipment.

9.2.3 Circuit Board

One nonstandard four-layer circuit board is used, measuring 3.1 inches x 2.8 inches. Figure 9-1 shows the layout of the DEC423 converter circuit board.

Figure 9-1: DEC423 Converter Circuit Board



MA-0944-87

9.2.4 Input/Output Connector Pinout

The input to the converter from the system module is through two D-sub connectors. Connector J5 (the 15-pin D-sub) accepts two of the three serial lines and three power supply voltages from the system module. Connector J4 (the 9-pin D-sub) accepts one serial line from the system module. Table 9-1 and Table 9-2 list J4 and J5 pinouts.

There are three MMJ connectors (J1, J2, and J3) at the output of the converter labeled 1 through 3, from left to right. The pin assignments on each MMJ connector are identical and Table 9-3 lists the pinout for all. The difference is that J1 is for the console terminal, J2 is the second (or auxiliary) terminal, and J3 is for the printer. However, J3 can be another terminal instead of a printer.

NOTE: *The VAXstation 2000 does not separate the signal and frame grounds of the RS232 ports. The metalized shell and all applicable ground pins on both D-sub connectors are tied to the same point. The converter will use the VAXstation 2000 ground for all its operation. Signal return currents and EOS/ESD currents are returned to the chassis through the D-sub shells and defined ground signal pins.*

Table 9-1: Connector J4 D-Sub Pinouts

Pin	Signal	Pin	Signal
1	Ground	6	No connection
2	SYS PTR_XDAT	7	Ground
3	SYS PTR_RDAT	8	No connection
4	No connection	9	No connection
5	No connection		

Table 9–2: Connector J5 D-sub Pinouts

Pin	Signal	Pin	Signal
1	No connection	9	No connection
2	No connection	10	No connection
3	No connection	11	No connection
4	+5 Vdc	12	-12 Vdc
5	SYS AUX_RDAT	13	SYS AUX_XDAT
6	Ground	14	SYS KBD_RDAT
7	Ground	15	SYS KBD_XDAT
8	+12 Vdc		

Table 9–3: MMJ Connector Pinouts for J1, J2, and J3

Pin	Signal	Pin	Signal
1	+5 Vdc	4	- Receive data
2	+ Transmit data	5	+ Receive data
3	- Transmit data	6	Buffered ground

9.2.5 Power Dissipation and Cooling

The total power dissipation of the converter assembly is 2.23 watts maximum, 1.94 watts typical. Most of the dissipation occurs within the three 9636 driver chips. Only one half of any driver chip is connected to the outside cables so as to spread the greatest power dissipation across all the chips. There are no louvers on the plastic housing, so the cooling process is one of thermal conduction from the chips to the multilayer PC board and the surrounding plastic, where the chips act as convection heat sinks to the local ambient temperature.

9.2.6 Power Supply

All the power used by the converter is supplied through the D-sub connectors by the system module. The current levels for each supply are listed below.

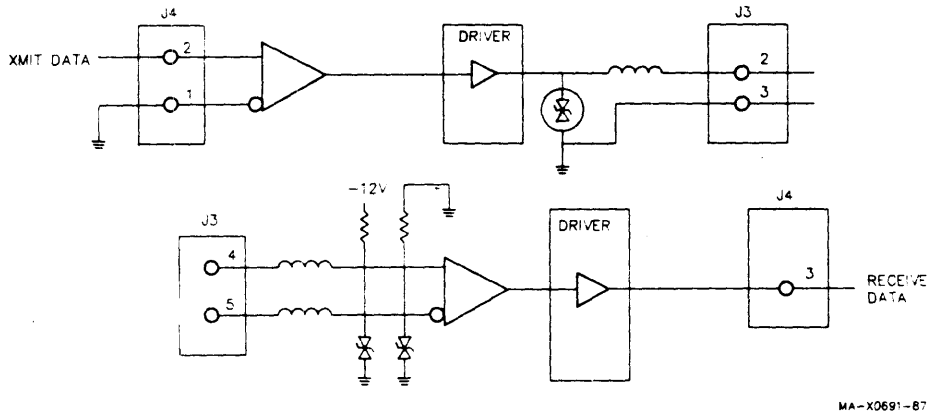
+12 Vdc 700 milliamp maximum
-12 Vdc 854 milliamp maximum
+5 Vdc 677 milliamp maximum

9.3 Circuit Descriptions

The system module does not contain the protection circuitry or the proper layout to conform to the requirements of DEC STD 52-4, even though they use DEC423 compatible drivers and receivers. Each of the three serial lines that come from the system module are first converted into TTL, and then are converted to DEC 423. These drivers/receivers reside as close as possible to the D-sub connectors. The remainder of the physical space between the drivers/receivers and the output MMJ connectors contains the protection circuits, line terminators, and failsafe components.

All three serial lines in the DEC423 converter are identical. The only difference is their connector pinouts. Figure 9–2 shows the serial line from the printer port, line 3.

Figure 9–2: DEC423 Converter Block Diagram for Line 3



9.3.1 Slew Rate

The DEC423 output driver circuits must interface with RS232 circuits through passive adapters. Such compatibility requires a slew rate resistor of 27K ohms for a risetime of 1.8 to 2.7 microseconds per DEC STD 52-4.

9.3.2 Failsafing

Per DEC STD 52-4, the 9639 receiver must be failsafed. That is, the input of the receivers must default to a predictable condition if they are disconnected from the terminal. Also, the input impedance of the 9639 receiver is not well-matched to RS232 and V.28 specifications. To meet these requirements, a 10K ohm resistor is connected from the positive side of the receiver to ground, and a 24K ohm resistor is connected between the negative side of the receiver and -12 V, as mandated by DEC STD 52-4. This will force the output of the receiver to the MARK condition if the cable is disconnected or if the terminal is powered off, keeping the system UART's inactive.

9.3.3 Pins 1 and 6 on the MMJ Connectors

Pins 1 and 6 of the MMJ connectors are unused within the converter. These lines are normally reserved for flow control signals in printers. When unused, DEC STD 52-4 requires that pin 1 be terminated with a 150 ohm resistor to +5 V. This line must also be protected from transients. Pin 6 must be terminated by a 3K ohm resistor to ground. Because of the larger impedance and the connection to ground, a transient supressor is not needed on line 6.

9.3.4 ESD/EOS Protection

All lines intended for external connection are protected with transient suppressors where necessary. All receiver lines are protected by an integrated package containing eight devices with a fusible link at a nominal voltage of 35 V. All driver lines are protected by discrete devices at a nominal voltage of 7 volts. Each of these parts are detailed in DEC STD 52-4. The converter passes all the tests required by DEC STD 52-4.

9.3.5 Chokes

Each of the active lines connected directly to a driver or receiver must have transient protection. The protection device is assisted by a 33 microHenry choke on each of these lines for two reasons.

1. The choke slows the leading edge of EOS or ESD. This gives the protection devices time to turn on and compensate for the lead and etch inductance of the protection device.
2. In those leads using discrete protection devices, the choke limits the current through the cable during a sustained high-current short by acting as a fuse. The fusing action of the choke is not needed on lines protected by the integrated protection chip. This chip has its own built-in fuse.

9.3.6 EMI/RFI Isolation and Susceptibility

The system box is not designed to operate with unshielded external data cables. The converter ensures that the MicroVAX 2000 system connects to the DTIS with FCC compliance using unshielded cables.

9.4 Loopback Connector H3103 (12-25083-01)

The loopback connector is a molded MMJ with the transmit and receive lines cross connected as shown below. These lines permit the looping of signals back to the system module to verify serial line operation.

Transmit data +/pin 2 ——>pin 5/Receive data +
Transmit data -/pin 3 ——>pin 4/Receive data -

Chapter 10
Expansion Peripherals

10.1 Introduction

This chapter describes the three expansion peripherals available with the VAXstation 2000 and MicroVAX 2000 systems. These are the hard disk expansion box, tape drive expansion box, and the expansion adapter. The hard disk expansion box is a mass storage device, the tape drive expansion box is removable tape cartridge mass storage device, and the expansion adapter interfaces both of the expansion boxes onto the system box.

10.1.1 Hard Disk Expansion Box

The hard disk expansion box contains a hard disk (Chapter 8), a power supply (Chapter 7), a resistor load module (Chapter 6), and the chassis. Since the drive, power supply, and the resistor load module are explained in other chapters, this section discusses the connector pinouts of the interface cable within the expansion box. Connector J1 is the 50-position D-sub connector which connects to the hard disk expansion box cable (BC17Y) from the system box. Connector J2 is the 34-position edge-card connector which connects to the rear of the disk drive. Connector J3 is the 20-position edge-card connector which connects to the rear of the disk drive. Table 10-1 lists the internal drive cable signals pinout.

Table 10–1: Hard Disk Expansion Box Internal Cable Pinout

J1	J2	Signal	J1	J3	Signal
17	34	Direction	1	20	Ground
18	33	Ground	2	19	Ground
19	32	Drive select 4	3	18	-Read data
20	31	Ground	4	17	+Read data
21	30	Drive select 3	5	16	Ground
22	29	Ground	6	15	Ground
23	28	No connection	7	14	-Write data
24	27	Ground	8	13	+Write data
25	26	No connection	9	12	Ground
26	25	Ground	10	11	Ground
27	24	Step	11	6	Ground
28	23	Ground	12	5	Spare
29	22	Ready	13	4	Ground
30	21	Ground	14	3	Reserved
31	20	index	15	2	Ground
32	19	Ground	16	1	Drive select acknowledge
33	18	Head select 1			
34	17	Ground			
35	16	No connection			
36	15	Ground			
37	14	Head select 0			
38	13	Ground			
39	12	Write fault			
40	11	Ground			
41	10	Track 0			
42	9	Ground			
43	8	Seek complete			

Table 10-1 (Cont.): Hard Disk Expansion Box Internal Cable Pinout

J1	J2	Signal	J1	J3	Signal
44	7	Ground			
45	6	Write gate			
46	5	Ground			
47	4	Head select 2			
48	3	Ground			
49	2	Head select 3			
50	1	No connection			

10.1.2 Tape Drive Expansion Box

The tape drive expansion box contains a TK50 tape drive (See Chapter 8), a TZK50 SCSI controller board (Refer to the *TZK50/SCSI Controller Technical Manual* order number EK-TZK50-TM), a power supply (See Chapter 7), a resistor load module (See Chapter 6), and the chassis. Since the tape drive, power supply, and the resistor load module are explained in other chapters and the TZK50 controller is explained in the above referenced document, this section discusses the connector pinout of the interface cable between the TZK50 controller board and the external connector. Table 10-2 lists the pinout signals for the internal tape drive. Connector J1 is the 50-position IEEE connector which connects to the tape expansion box cable (BC19J) from the system box. Connector J2 is the 50-position edge-plug connector which connects to the SCSI port on the TZK50 controller board. There is another connector on this cable (J3) and it has a one-to-one pinout with the J1 connector. Connector J3 is used for daisy chaining expansion boxes. Although the VMS and ULTRIX operating system software do not support more than one tape expansion box, future operating systems that do support multiple tape expansion boxes will use the J3 connector for daisy chaining.

Table 10–2: Tape Drive Expansion Box Internal Cable Pinout

J1	J2	Signal	J1	J2	Signal
1	1	Ground	26	2	Termination power
2	3	Data bus 0	27	4	Ground
3	5	Ground	28	6	No connection
4	7	Data bus 1	29	8	Ground
5	9	Ground	30	10	No connection
6	11	Data bus 2	31	12	Ground
7	13	Ground	32	14	Attention
8	15	Data bus 3	33	16	Ground
9	17	Ground	34	18	No connection
10	19	Data bus 4	35	20	Ground
11	21	Ground	36	22	Busy
12	23	Data bus 5	37	24	Ground
13	25	Ground	38	26	Acknowledge
14	27	Data bus 6	39	28	Ground
15	29	Ground	40	30	Reset
16	31	Data bus 7	41	32	Ground
17	33	Ground	42	34	Message
18	35	Data bus parity	43	36	Ground
19	37	Ground	44	38	Select
20	39	No connection	45	40	Ground
21	41	Ground	46	42	Command/Data
22	43	No connection	47	44	Ground
23	45	Ground	48	46	Request
24	47	No connection	49	48	Ground
25	49	No connection	50	50	Input/Output

10.1.3 Expansion Adapter

The expansion adapter attaches to the bottom of the system box and is basically a cable interface device. It has three ports for external devices. Port A is for the tape drive expansion box, port B is for the hard disk expansion box, and port C is for the serial line unit options on the MicroVAX 2000 system. Port C is reserved for future options on the VAXstation 2000 system.

10.1.3.1 The Tape Port (Port A)

Table 10-3 lists the internal cables signal pinout on port A which interfaces the tape drive expansion box to the 5380 tape controller chip. Connector J1 is the 50-position IEEE connector which is port A on the expansion adapter. Connector J2 is the 50-position berg connector which connects to the tape port on the system module.

Table 10-3: Tape Port Internal Cable Pinout (Port A)

J1	J2	Signal	J1	J2	Signal
1	1	Ground	26	38	No connection
2	26	DBUS0	27	14	Ground
3	2	Ground	28	39	Ground
4	27	DBUS1	29	15	Ground
5	3	Ground	30	40	Ground
6	28	DBUS2	31	16	Ground
7	4	Ground	32	41	SCATN
8	29	DBUS3	33	17	Ground
9	5	Ground	34	42	Ground
10	30	DBUS4	35	18	Ground
11	6	Ground	36	43	SCBSY
12	31	DBUS5	37	19	Ground
13	7	Ground	38	44	SCACK
14	32	DBUS6	39	20	Ground
15	8	Ground	40	45	SCRST
16	33	DBUS7	41	21	Ground
17	9	Ground	42	46	SCMSG

Table 10–3 (Cont.): Tape Port Internal Cable Pinout (Port A)

J1	J2	Signal	J1	J2	Signal
18	34	DBUSP	43	22	Ground
19	10	Ground	44	47	SCSEL
20	35	Ground	45	23	Ground
21	11	Ground	46	48	SCC/D
22	36	Ground	47	24	Ground
23	12	Ground	48	49	SCREQ
24	37	Ground	49	25	Ground
25	13	No connection	50	50	SCI/O

10.1.3.2 The Disk Port (Port B)

The disk port (port B) on the expansion adapter has a disk interface module attached to it. This module converts two berg-style connectors from the system module that have the disk data bus on them into a single 50-position D-sub connector for connection to the hard disk expansion box. The hard disk expansion box is connected to port B via the disk expansion box cable (BC17Y). Table 10–4 lists the disk interface module pinout that interfaces the hard disk expansion box to the 9224 disk controller chip through port B of the expansion adapter. Connector J1 is the 26-position connector which contains the disk control signals from the system module. Connector J2 is the 20-position connector which contains the read and write data from the system module. Connector J3 is the 50-position D-sub connector which is port B on the expansion adapter.

Table 10–4: Disk Interface Module Pinout (Port B)

J1	J3	Signal	J2	J3	Signals
1	17	Head select 3	1	6	Drive select acknowledge
2	49	Head select 2	2		Ground
3		Ground	3	38	Reserved
4	32	Write gate	4		Ground
5	15	Seek complete	5	21	Spare
6		Ground	6		Ground
7	47	Track 0	7		No connection
8	30	Write fault	8		No connection
9		Ground	9		No connection
10	13	Head select 0	10		No connection
11	28	Head select 1	11		Ground
12		Ground	12		Ground
13	11	Index	13	36	+Write data
14	43	Ready	14	3	-Write data
15		Ground	15		Ground
16	26	Step	16		Ground
17	7	Drive select 4	17	2	+Read data
18		Ground	18	18	-Read data
19	24	Drive select 3	19		Ground
20		Ground	20		No connection
21	39	Direction			
22		No connection			
23		No connection			
24		No connection			
25		No connection			
26		No connection			

NOTE: *All pins not listed for J3 are connected to ground with the exception of pins 9, 41, and 45, which have no connection.*

Appendix A
Timing Diagrams

This appendix contains sample timing diagrams.

Figure A-1: DAL Bus Address Control

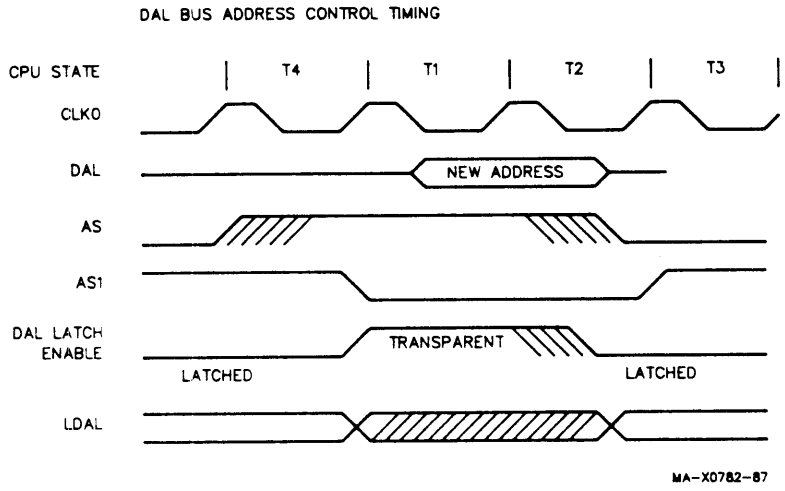
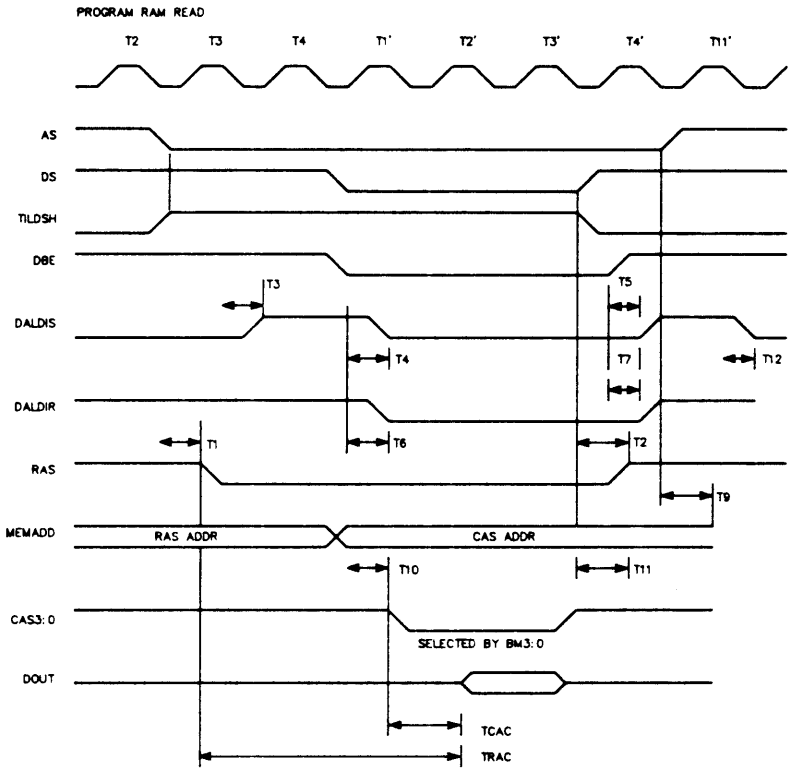


Figure A-2: Program RAM Read

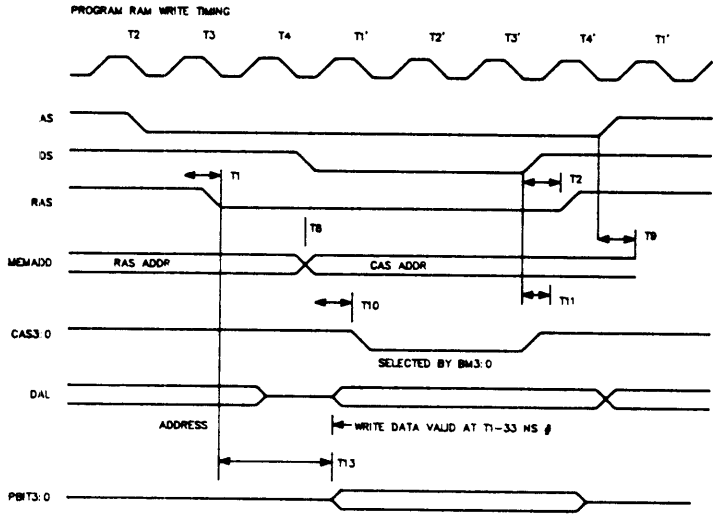


PARAMETER	DESCRIPTION	TIME	TIME FROM T3
T1	SRAS (H->L) FROM T3	36	36
T2	SRAS (L->H) FROM DS (L->H)	38	N/A
T3	DALDIS (L->H) FROM T3	42	42
T4	DALDIS (L->H) FROM DBE (H->L)	16	N/A
T5	DALDIS (L->H) FROM DBE (L->H)	14	N/A
T6	DALDIR (H->L) FROM DBE (H->L)	27	N/A
T7	DALDIR (L->H) FROM DBE (L->H)		
T8	MEMADD RAS->CAS ADDRESS FROM T4	43	93
T9	MEMADD CAS DEASSERT FROM AS (L->H)		
T10	CAS (H->L) FROM T4.5	37	92
T11	CAS (L->H) FROM DS (L->H)	40	N/A
T12	DALDIS (H->L) FROM T1*		
TCAC	CAS ACCESS TIME	>75 *	
TRAC	RAS ACCESS TIME	>150 *	

* DEPENDENT ON MEMORY BUFFERS.
NOTE THAT ALL TIMES ARE PRELIMINARY AND SUBJECT TO CHANGE.
TILDSH IS AN INTERNAL SIGNAL WHICH ENABLES RAS AND CAS FROM THE FALLING EDGE OF AS TO THE RISING EDGE OF DS.

MA-X07783-87

Figure A-3: Program RAM Write



T1 SRAS (H->L) FROM T3	36	36
T2 SRAS (L->H) FROM DS (L->H)	38	N/A
T8 MEMADD RAS->CAS ADDRESS FROM T4	43	93
T9 MEMADD CAS DEASSERT FROM AS (L->H)		
T10 CAS (H->L) FROM T1'	37	92
T11 CAS (L->H) FROM DS (L->H)	40	N/A
T13 PARITY DATA VALID AFTER RAS		

NOTES - CAS STARTS ONE HALF CLK0 PERIOD LATER THAN FOR READ CYCLE TO ALLOW ADDITIONAL TIME TO CALCULATE PARITY.
- DALDIR REMAINS HIGH FOR ENTIRE CYCLE.
- DALDIS REMAINS LOW FOR ENTIRE CYCLE.
MEASURED AT THE UVAX CPU CHIP.
ALL TIMES ARE PRELIMINARY AND SUBJECT TO CHANGE.

MA-X07B4-87

Figure A-4: I/O Single Cycle Read

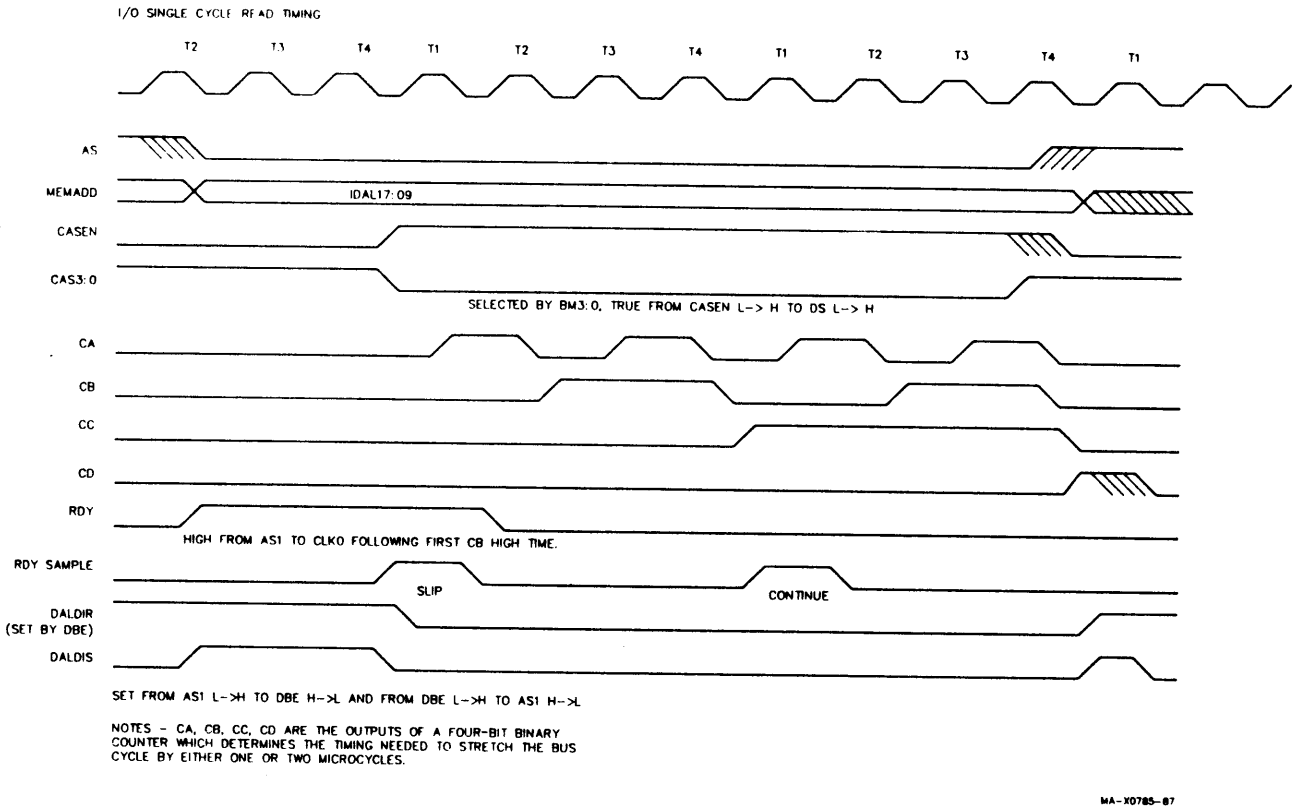


Figure A-5: I/O Single Cycle Write

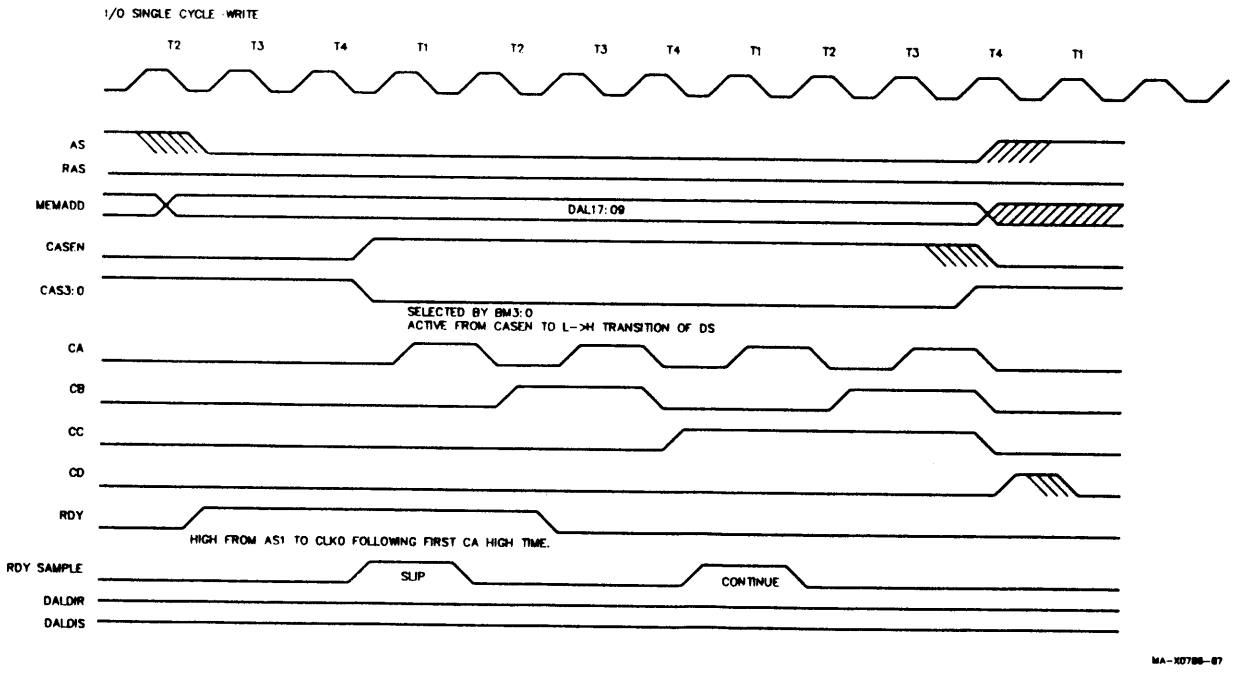


Figure A-6: Video RAM Read

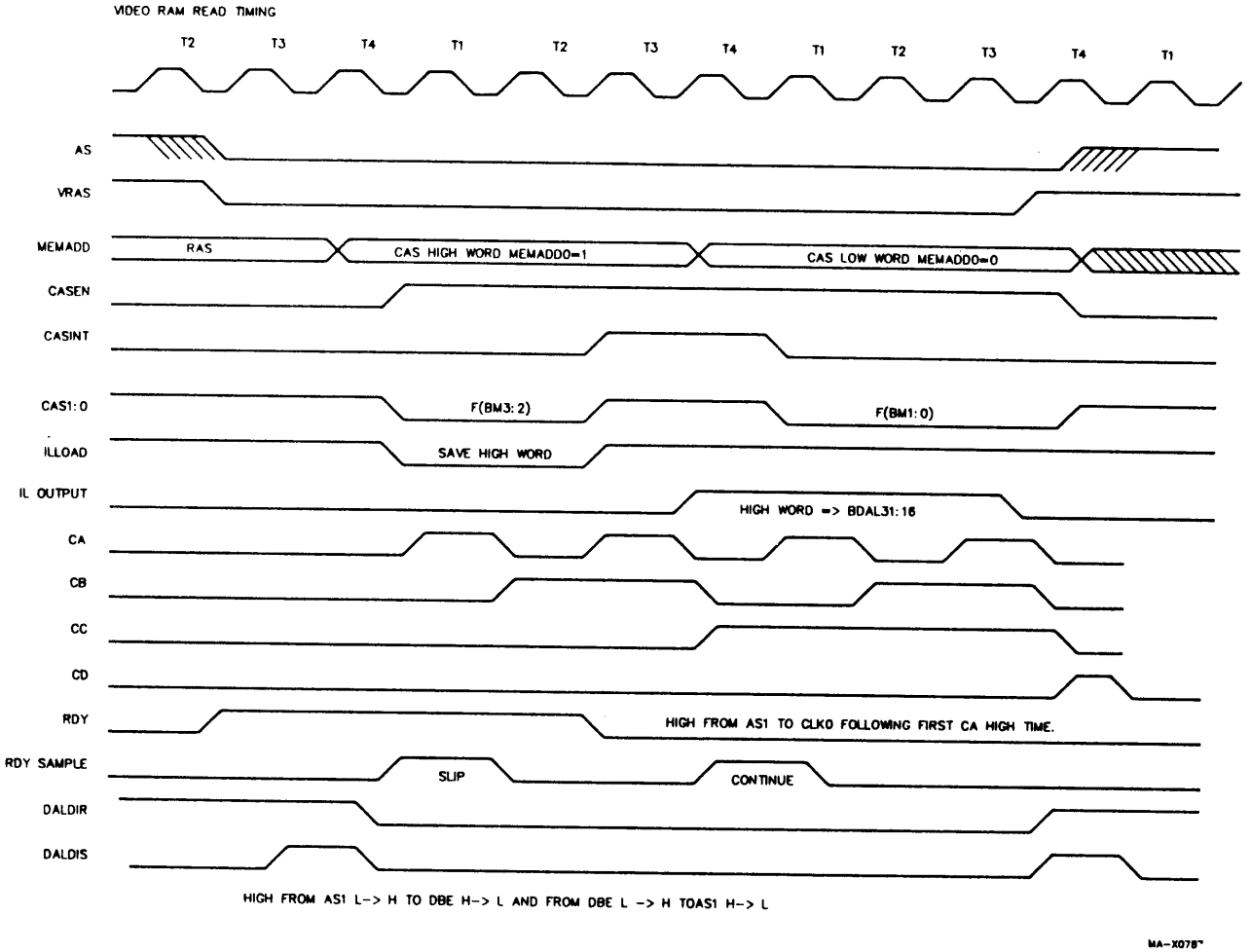


Figure A-7: Video RAM Write

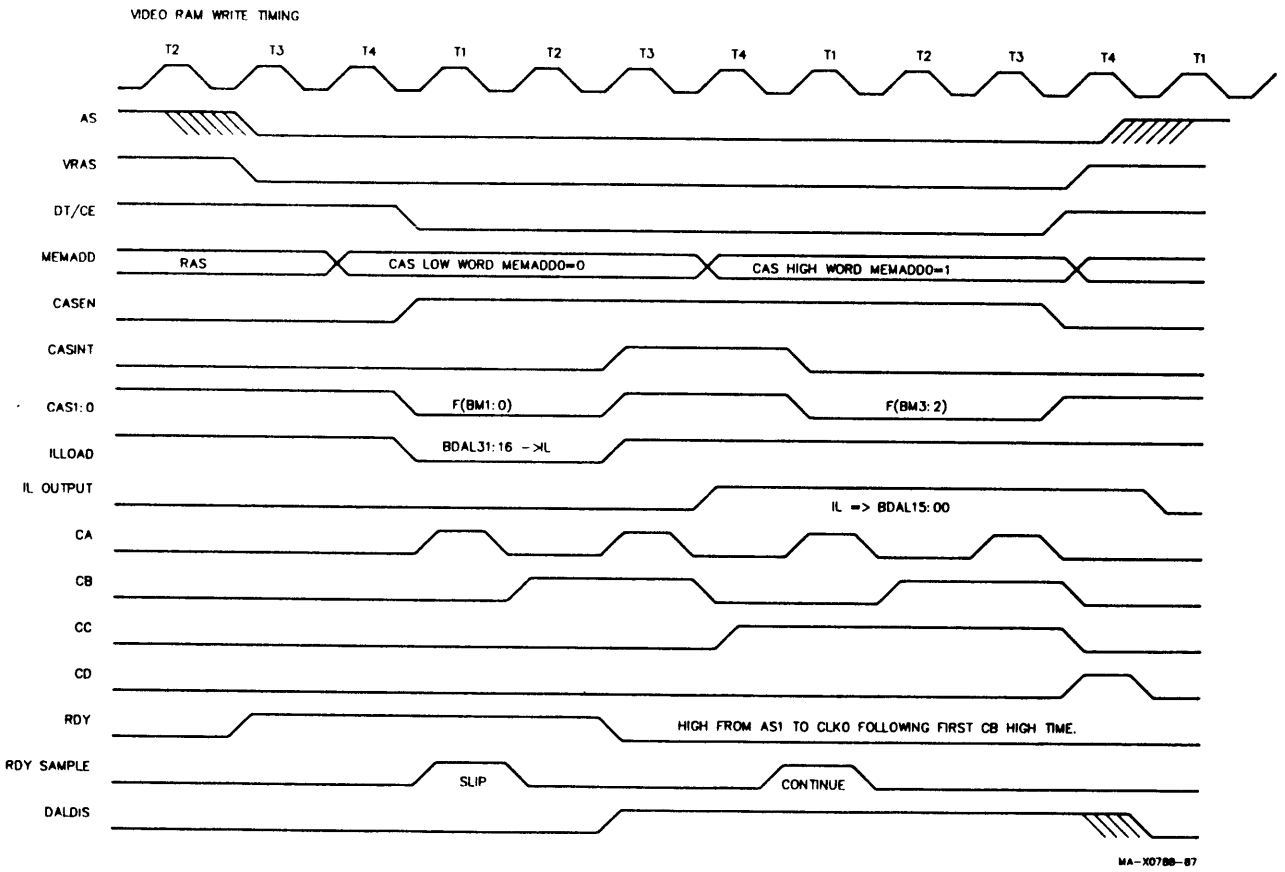


Figure A-8: I/O Double Cycle Read

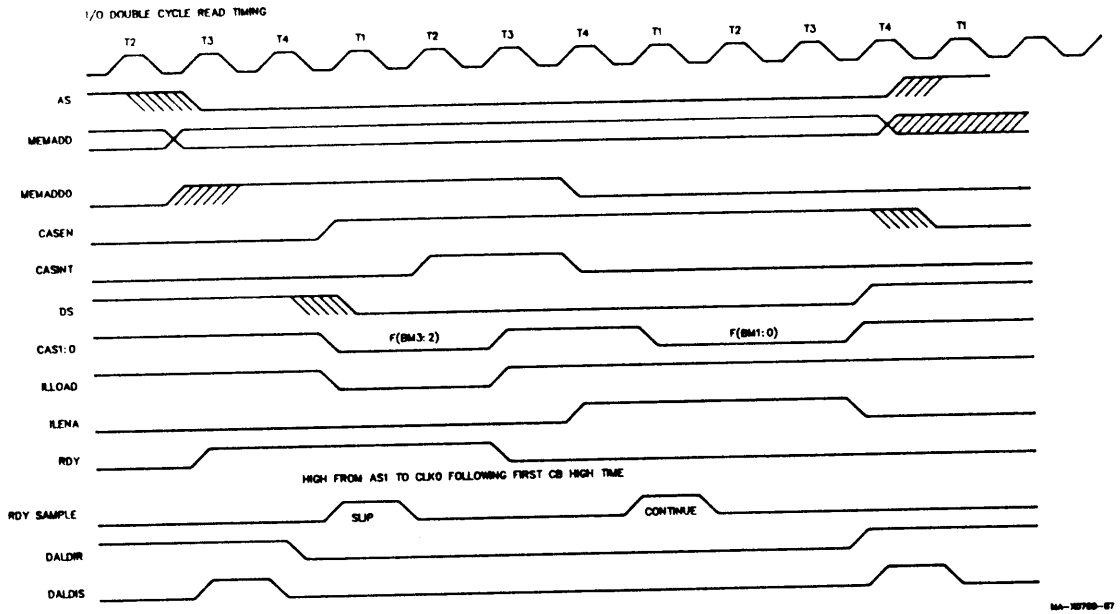


Figure A-9: I/O Double Cycle Write

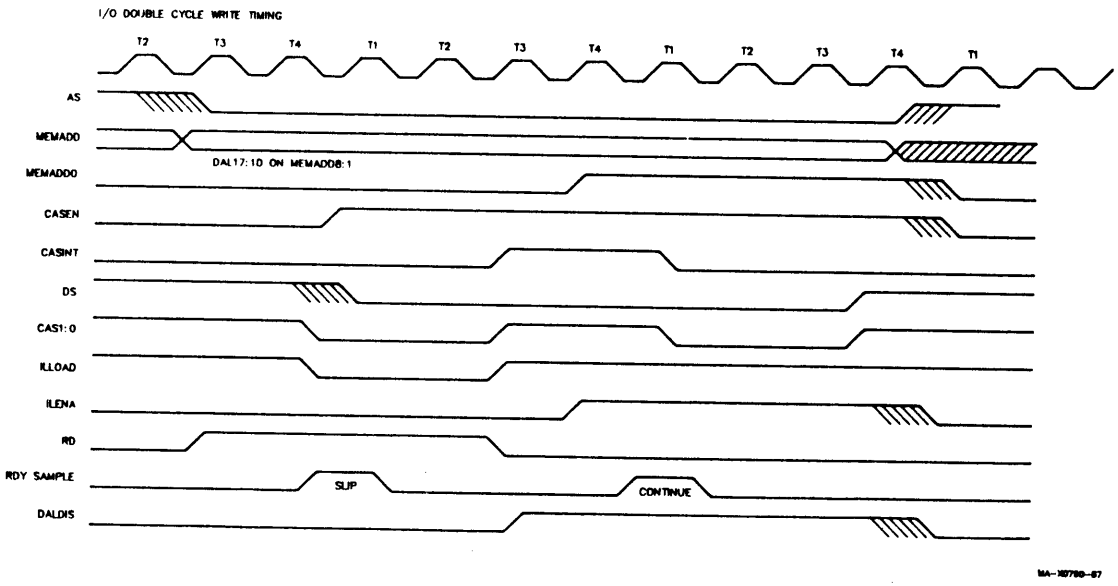


Figure A-10: CPU Cycle Slips

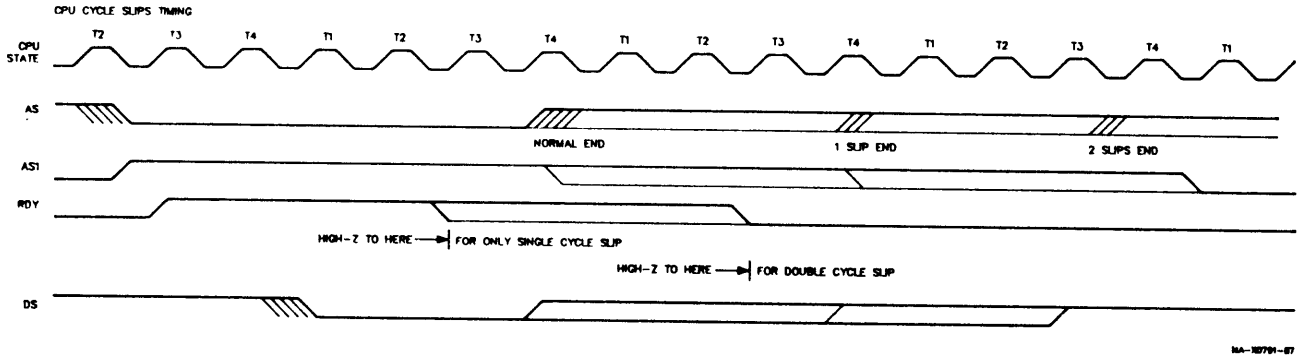


Figure A–11: Video Shift Register Update Cycle

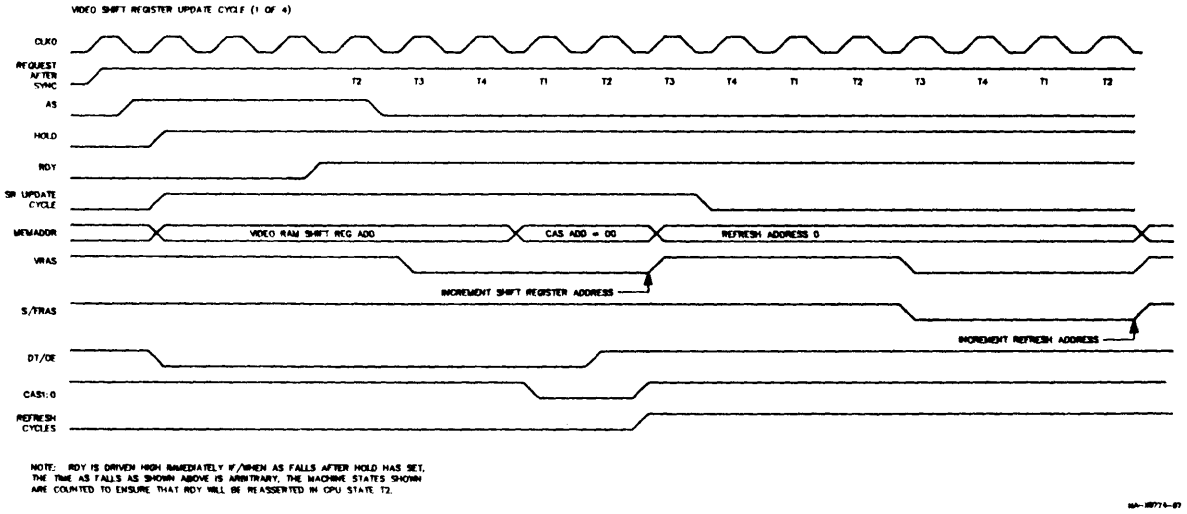


Figure A–11 (Cont.): Video Shift Register Update Cycle

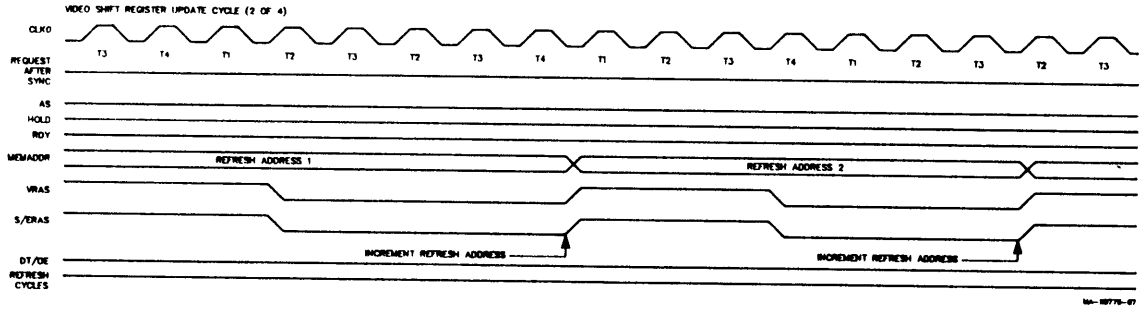


Figure A–11 (Cont.): Video Shift Register Update Cycle

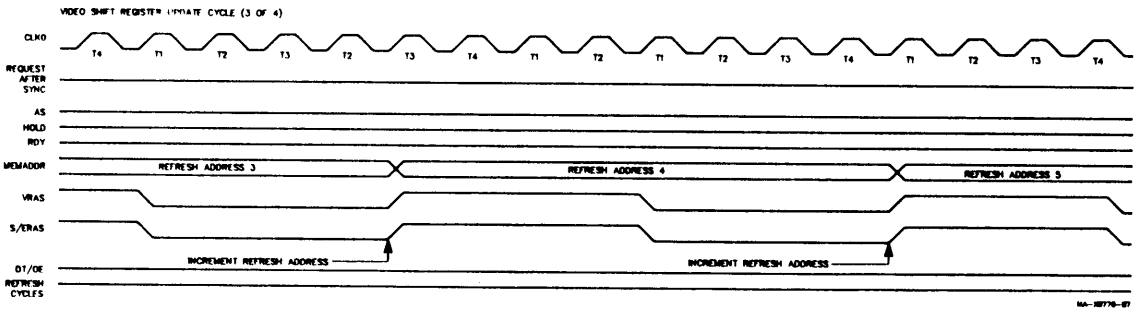


Figure A-11 (Cont.): Video Shift Register Update Cycle

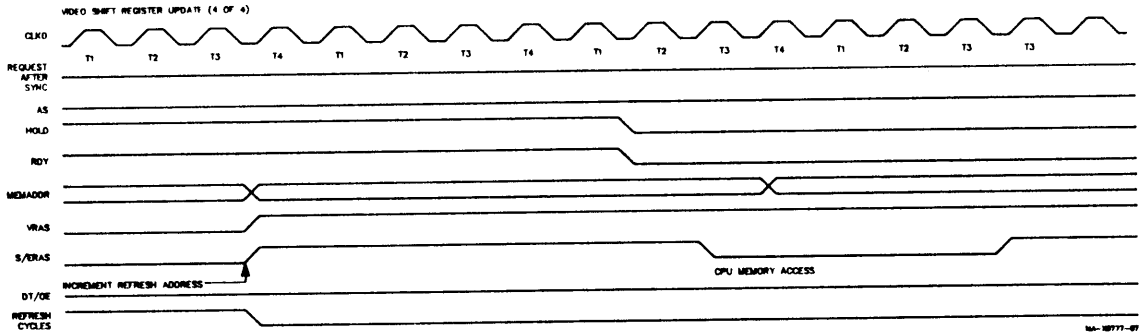


Figure A–12: Start of Display/Region Line

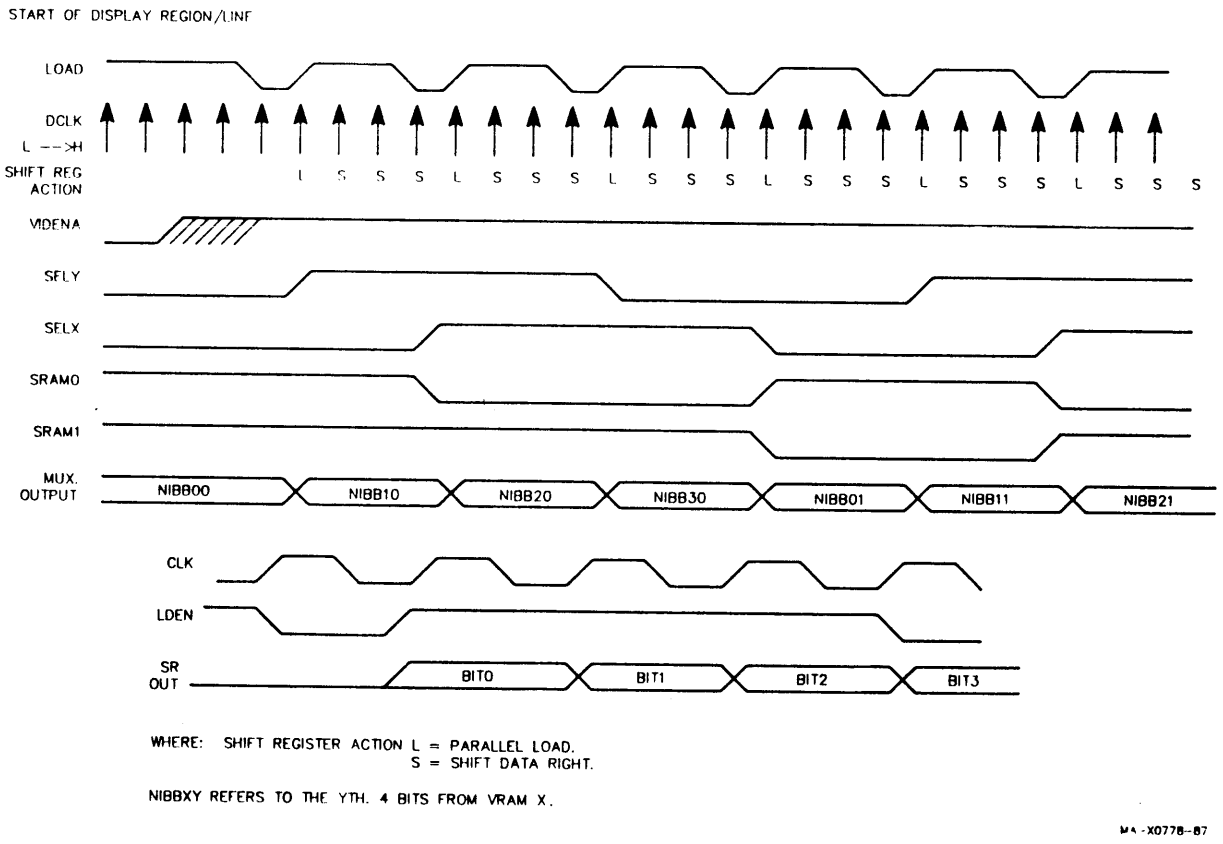


Figure A-13: End of Display/Region Line

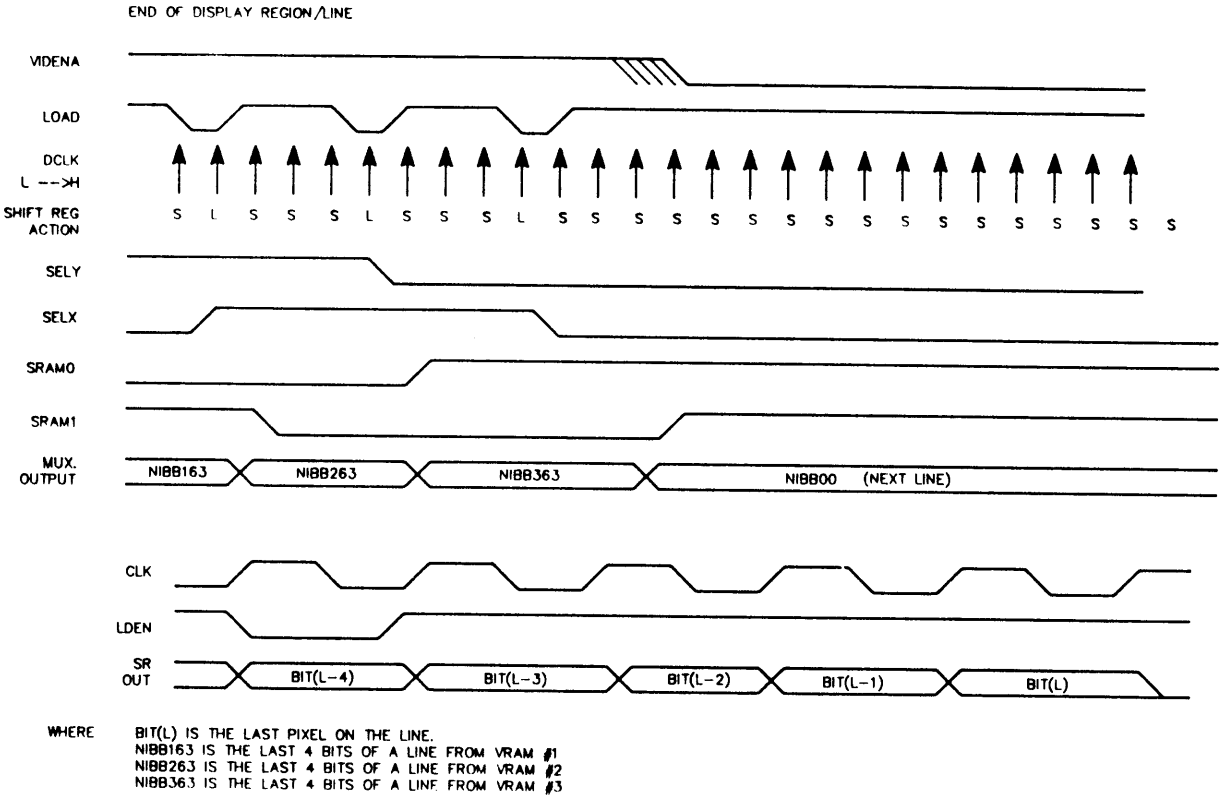
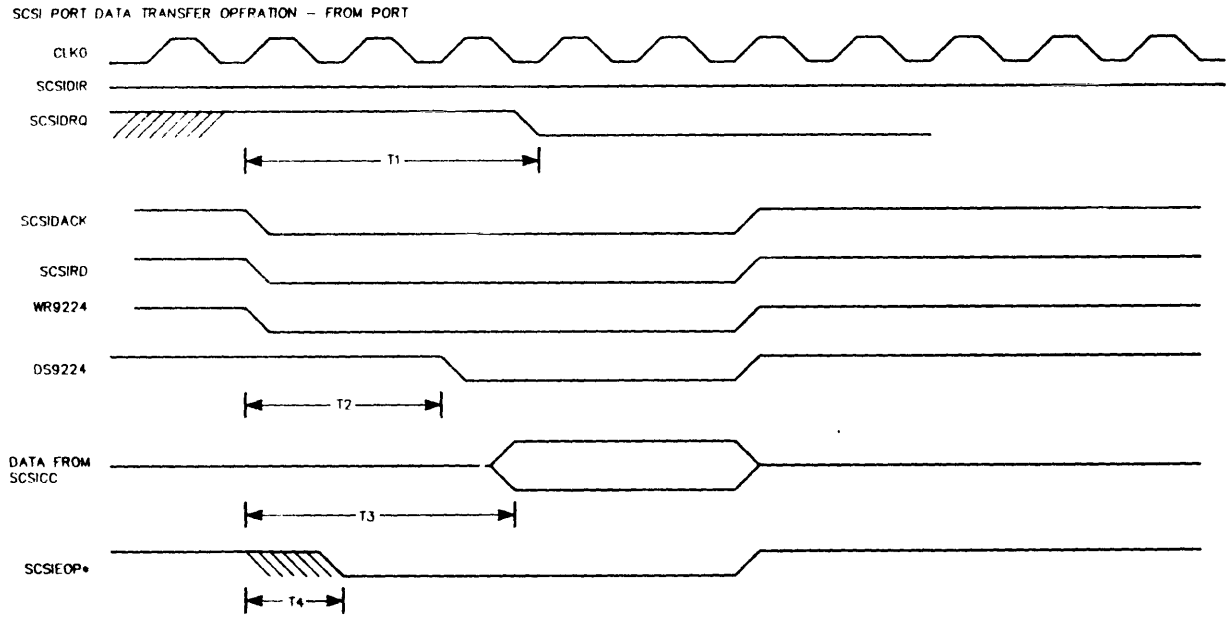


Figure A-14: Tape (SCSI) Port Data Transfer Operation (From Port)

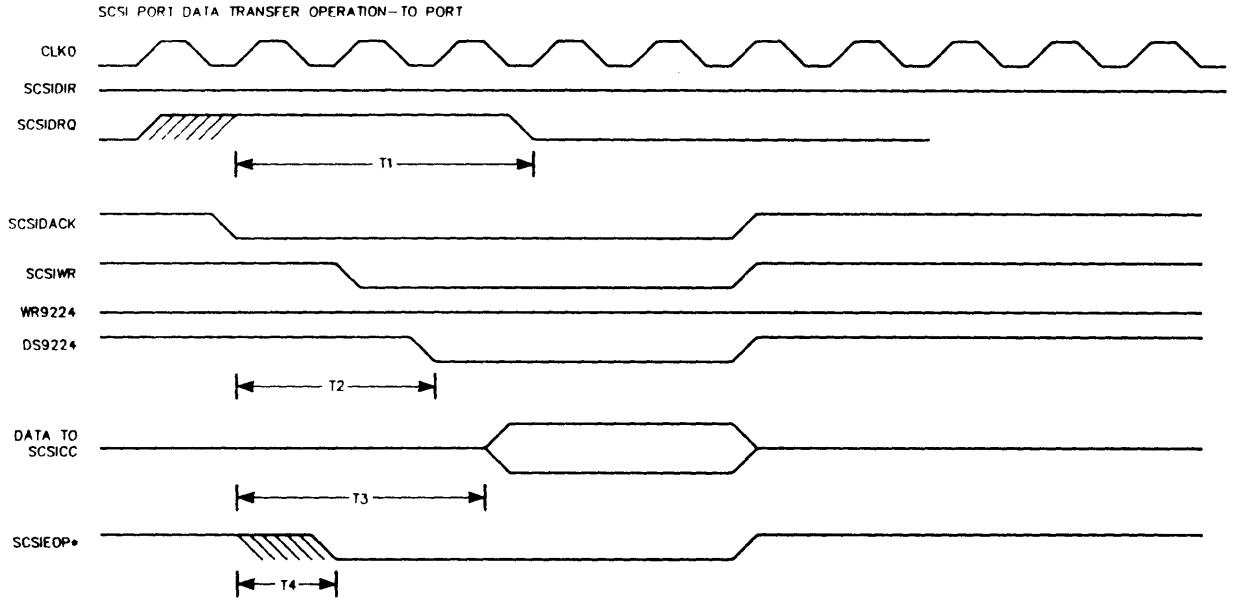


* ONLY ASSERTED IF SCSI BYTE COUNT REGISTER BECOMES 0000H AS A RESULT OF THIS SCSIDRQ.

PARAMETER	MIN.	MAX	NOTES
T1	–	130	PARAMETER OF SCSI CONTROLLER CHIP
T2	–		
T3	–	135	PARAMETER OF SCSI CONTROLLER CHIP
T4	–		BYTE COUNT REGISTER RIPPLE CARRY TIME

MA-X0773-87

Figure A-15: Tape (SCSI) Port Data Transfer Operation (To Port)



* ONLY ASSERTED IF SCSI BYTE COUNT REGISTER BECOMES 0000H AS A RESULT OF THIS SCSIDRQ.

PARAMETER	MIN.	MAX	NOTES
T1	–	130	PARAMETER OF SCSI CONTROLLER CHIP
T2	–		
T3	–	135	PARAMETER OF SCSI CONTROLLER CHIP
T4	–		BYTE COUNT REGISTER RIPPLE CARRY TIME

MA-X0772-87

Appendix B
Physical Address Maps

B.1 System Module Addresses

The addresses used by hardware on the KA410 system module and the MS400 RAM memory option module are listed in Table B-1.

Table B-1: System Module Address Locations

Address Range	Symbolic Name	Description
0000.0000-001F.FFFF		System module RAM
0020.0000-00FF.FFFF		Memory option module RAM
2002.0000	CFGTST	Configuration & test register (r/o)
2002.0000	IORESET	I/O reset register (w/o)
2004.0000-2007.FFFF		System module ROM (up to 256 kilobytes)
2004.0004	SYS_TYPE	System ID extension register
2004.0020-2004.003F		Interrupt vector numbers
2008.0000	HLTCOD	Halt code register
2008.0004	MSER	Memory system error register
2008.0008	MEAR	Memory error address register
2008.000C	INT_MSK	Interrupt mask register
2008.000D	VDC_ORG	Monochrome display origin
2008.000E	VDC_SEL	Video interrupt select
2008.000F	INT_REQ	Interrupt request register (r/o)
2008.000F	INT_CLR	Interrupt request clear (w/o)

Table B-1 (Cont.): System Module Address Locations

Address Range	Symbolic Name	Description
2009.0000-2009.007F		Network address ROM
200A.0000-200A.000F	SER_xxx	Serial line controller
200B.0000-200B.00FF	WAT_xxx	Time-of-year clock and NV RAM
200C.0000-200C.0007	DKC_xxx	Disk controller ports
200C.0080-200C.009F	SCS_xxx	Tape (SCSI) controller chip
200C.00A0	SCD_ADR	Tape (SCSI) DMA address register
200C.00C0	SCD_CNT	Tape (SCSI) DMA byte count register
200C.00C4	SCD_DIR	Tape (SCSI) DMA transfer direction
200D.0000-200D.3FFF		Disk data buffer RAM
200F.0000-200F.003F	CUR_xxx	Monochrome video cursor chip
3000.0000-3001.FFFF	Monochrome video RAM	

B.2 Option Module Address Ranges

The following address ranges are defined for use by option modules connected to the network option and video option connectors. For some of these ranges hardware on the system module generates a selection signal, whose name is listed. If no signal name is listed, the option module must decode the address range from the data/address bus. Table B-2 lists the nominal ranges. Subsequent tables show the actual ranges used by each option type.

Table B-2: Option Module Address Ranges

Address Range	Description
200E.0000-200E.FFFF	Network option, signal NIENA
2200.0000-23FF.FFFF	Future option CSRs
2400.0000-25FF.FFFF	Future option CSRs
2010.0000-2013.FFFF	Network option ROM, signal NIROMCS
2014.0000-2017.FFFF	Video option ROM, signal OPTROMENA
2018.0000-201B.FFFF	Additional option 1 ROM
201C.0000-201F.FFFF	Additional option 2 ROM
3800.0000-3BFF.FFFF	Video option (32-bit path), signal OPTVIDENA
3C00.0000-3C00.FFFF	Video option (16-bit path), signal OPTVIDENA

B.2.1 Ethernet Network Option Addresses

Table B-3 shows the addresses used by the DESVA Ethernet network option that is described in chapter 5.

Table B-3: Ethernet Network Option Module Addresses

Address Range	Description
200E.0000-200E.0007	LANCE chip registers
2010.0000-2011.FFFF	Firmware ROM (one 32 Kbyte chip)

B.2.2 Graphics (Color) Video Option Addresses

Table B-4 shows the addresses used by the graphics video option.

Table B-4: Graphics Video Option Module Addresses

Address Range	Description
2014.0000-2015.FFFF	Firmware ROM (one 32 kilobyte chip)
3C00.0000-3C00.007F	ADDER chip registers
3C00.0200-3C00.02FF	FIFO compression chip registers
3C00.0300-3C00.037F	Video DAC registers
3C00.0400-3C00.041F	Cursor chip registers
3C00.0500-3C00.0501	Video readback register
3C00.8000-3C00.FFFF	FIFO/template RAM

B.2.3 Eight-port Asynchronous Serial Line Addresses

Table B-5 shows the addresses by the 8-port asynchronous serial line option which is installed in the graphics option port.

Table B-5: Asynchronous SLU Option Module Addresses

Address Range	Description
2014.0000-2015.FFFF	Firmware ROM (one 32 kilobyte chip)
3800.0000-3800.000F	Control and status registers

Index

A

B

C

D

E

F

H

R

S

T

V

W

TECHNICAL DOCUMENTATION
CHANGE NOTICE

EK-VTTAA-TM-CN2

VAXstation 2000 and MicroVAX 2000 TECHNICAL MANUAL

EK-VTTAA-TM-001

This notice contains a change to the *VAXstation 2000 and MicroVAX 2000 Technical Manual.*
Update your manual as indicated with the information presented below.

PAGE 3-138

Change the description of the Data Rate (Bits/Second), as follows.

FROM:

11	10	9	8	Data Rate (Bits/Seconds)
1	1	1	1	19200

TO:

11	10	9	8	Data Rate (Bits/Seconds)
1	1	1	1	19800 (Nonstandard)

digital equipment corporation • maynard, massachusetts

TECHNICAL DOCUMENTATION CHANGE NOTICE

EK-VTTAA-CN-001

This new Chapter 4, MS400 Option Memory Modules, replaces the existing Chapter 4 in the VAXstation 2000 and MicroVAX 2000 Technical Manual, EK-VTTAA-TM-001.

digital equipment corporation • maynard, massachusetts

Chapter 4
MS400 Option Memory Modules

4.1 Introduction

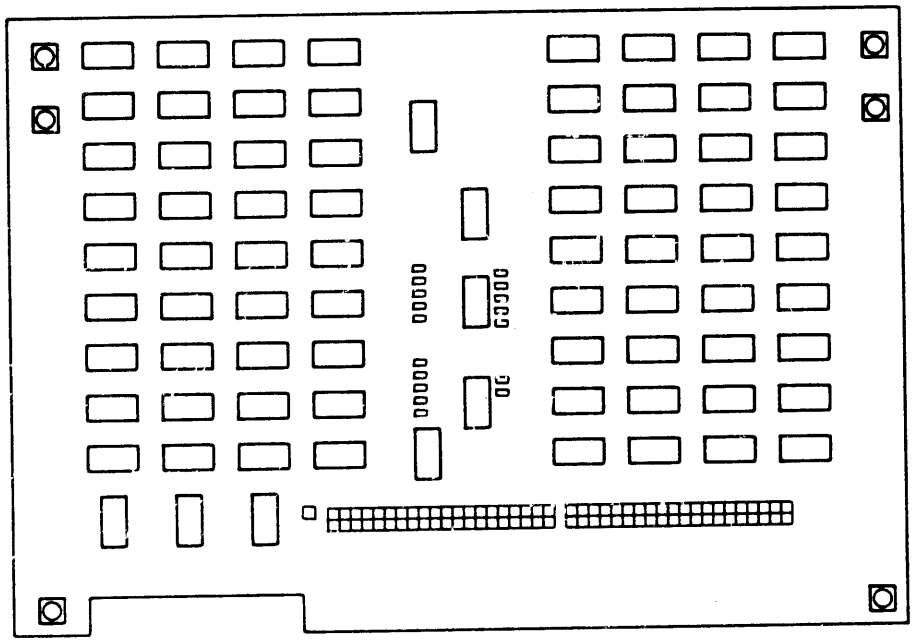
This chapter describes the MS400-AA, MS400-BA, and MS400-CA memory modules that are options to the KA410-AA system module. The MS400-AA memory module contains 2 megabytes of memory, the MS400-BA memory module contains 4 megabytes of memory, and the MS400-CA memory module contains 12 megabytes of memory. The MS400-BA and MS400-CA have components on both sides of the module. Both the MS400-AA and MS400-BA utilize the 256K DRAMs while the MS400-CA utilizes the 1M DRAMs. Only one MS400 memory module may be connected to a KA410-AA system module. Figure 4-1 shows a front view of the MS400 series memory module.

These MS400 series modules do not provide RAM control signal generation; however, they do provide transceivers for data and buffers for driving the RAM array with RAS, CAS, WRITE, and ADDRESS. The KA410-AA system module generates byte parity when writing to RAM memory and checks byte parity when reading from RAM memory. Parity checking applies both to CPU accesses and to DMA accesses generated by the network controller option. Only those bytes selected by the processor byte mask are affected and checked.

4.2 Theory of Operation

MS400 option memory is contained in DRAMs. These are the same DRAMs as described in Section 3.3.1.1. The control signals on the memory module and the timing cycles are described in this section.

Figure 4-1: MS400 Memory Module



MAX-0882-87

4.2.1 Memory Module Control Signal Descriptions

Signal ERAS L is the RAS timing signal for the memory on the option module. ERAS is asserted for normal read and write cycles on the memory module (such as physical addresses in the range 0020.0000 through 00FF.FFFF). Signal SRAS L is the RAS timing signal for RAM memory on the base system module (physical addresses in the range 0000.0000 through 001F.FFFF). SRAS is negated during normal read and write cycles on the memory module. During refresh cycles, both ERAS and SRAS are asserted.

Bits 22, 21 and 20 of the system data/address bus (BDAL22, BDAL21, and BDAL20 on the system module that map to MSEL22, MSEL21, and MSEL20, respectively on the memory module) are latched in an F373 latch on the falling edge of VAS L. These latched address bits are decoded by an F138 which generates RAS for one of the four (or two) 1-megabyte memory arrays on the module. The appropriate decoder output is gated by ERAS true and SRAS false during normal read and write cycles and is input to the DRAM chip's RAS pins.

During a refresh cycle, both ERAS and SRAS are asserted. This negates all the outputs of the decoders and switches the multiplexors to assert RAS to all the DRAM chips on the option module.

The four CASx L signals from the system module pass through F244 buffers and series damping resistors to the CAS pins on the DRAM chips. Each CAS signal is associated with one of the processor byte masks and so determines which bytes of a longword are affected by a memory read or write cycle.

The multiplexed address lines MEMADDx H from the system module pass through F244 buffers and series damping resistors to the address pins on the DRAM chips. The timing of row address, RAS assertion, column address, and CAS assertion are controlled by the system module.

Signal BWRITE L from the system module passes through F244 buffers to the WE pins on the DRAM chips. This signal also controls the signal flow direction in the F245 data tranceivers.

The data input (D) and output (Q) pins of each DRAM chip are wired together and are sent to the system module data/address bus through F245 transceivers. The transceivers are enabled when both ERAS L and VDBE L are asserted. The direction of data flow is selected by the BWRITE L signal.

4.2.2 Memory Cycles

The memory module responds to three types of memory cycles. They are the read, write, and refresh cycles. Each cycle on the module is initiated by the assertion of ERAS L. The cycle type is determined by SRAS L and BWRITE L as shown in Table 4-1. The timing cycles for the memory module are described in Section 3.5.2.

Table 4-1: Determining Memory Cycles

Cycle Type	ERAS L	SRAS L	BWRITE L
Read	True	False	False
Write	True	False	True
Refresh	True	True	False

4.3 Connector Pinouts

Connector J1 carries power, address, and control signals as listed in Table 4-2. Connector J2 carries the buffered processor data/address bus (BDAL31:00) as listed in Table 4-3.

Table 4-2: Connector J1 Pinout

Pin	Signal	Description
1	+5 VC	
2	+5 VB	
3	GND	
4	GND	
5	PBIT03 H	Parity bit for byte 3
6	PBIT02 H	Parity bit for byte 2
7	PBIT01 H	Parity bit for byte 1
8	PBIT00 H	Parity bit for byte 0
9	MSIZE2 L	Memory size bit 2
10	MEMAD8 H	Multiplexed address bit 8
11	MEMAD7 H	Multiplexed address bit 7
12	MEMAD6 H	Multiplexed address bit 6
13	GND	
14	GND	
15	MEMAD5 H	Multiplexed address bit 5
16	MEMAD4 H	Multiplexed address bit 4
17	MEMAD3 H	Multiplexed address bit 3
18	MEMAD2 H	Multiplexed address bit 2
19	MEMAD1 H	Multiplexed address bit 1
20	MEMAD0 H	Multiplexed address bit 0
21	MSIZE1 L	Memory size bit 1
22	MSIZE0 L	Memory size bit 0
23	CAS3 L	CAS for byte 3
24	CAS2 L	CAS for byte 2

Table 4–2 (Cont.): Connector J1 Pinout

Pin	Signal	Description
25	CAS1 L	CAS for byte 1
26	CAS0 L	CAS for byte 0
27	GND	
28	GND	
29	MSELC H	BDAL22 H from system
30	ERAS L	Extended RAS (ERAS from the standard cell)
31	SRAS L	Standard RAS (SRAS0 from the standard cell)
32	MSELB H	BDAL21 H from system
33	MSELA H	BDAL20 H from system
34	VAS L	Address strobe (BAS1 L on system module)
35	VDBE L	Data bus enable
36	BWRITE L	Write (BWRITE1 L on system module)
37	GND	
38	GND	
39	+5 VA	
40	+5 VA	

Table 4-3: Connector J2 Pinout

Pin	Signal	Pin	Signal
1	GND	21	BDAL15 H
2	GND	22	BDAL14 H
3	BDAL31 H	23	BDAL13 H
4	BDAL30 H	24	BDAL12 H
5	BDAL29 H	25	BDAL11 H
6	BDAL28 H	26	BDAL10 H
7	BDAL27 H	27	GND
8	BDAL26 H	28	GND
9	BDAL25 H	29	BDAL09 H
10	BDAL24 H	30	BDAL08 H
11	BDAL23 H	31	BDAL07 H
12	BDAL22 H	32	BDAL06 H
13	GND	33	BDAL05 H
14	GND	34	BDAL04 H
15	BDAL21 H	35	BDAL03 H
16	BDAL20 H	36	BDAL02 H
17	BDAL19 H	37	BDAL01 H
18	BDAL18 H	38	BDAL00 H
19	BDAL17 H	39	GND
20	BDAL16 H	40	GND

4.4 Configuration Jumpers

There are no field-modifiable jumpers on the module. The version of the module is determined by three signals on connector J1. These three signals are either disconnected (open) or grounded to indicate which memory module is installed. Table 4-4 lists the three signals and the preset configuration jumpers for each memory module.

Table 4–4: Memory Module Configuration Jumpers

Signal	Pin (J1)	MS400-AA	MS400-BA	MS400-CA
MSIZE2 L	9	Open	Open	Ground
MSIZE1 L	21	Ground	Ground	Ground
MSIZE0 L	22	Open	Ground	Open

4.5 Power Requirements

The memory modules require +5 volts DC with a tolerance of plus or minus five percent. The typical current drawn is .5 amps.

VAXstation 2000
and MicroVAX 2000
Technical Manual
EK-VTTAA-TM-001

READER'S COMMENTS

Your comments and suggestions will help us in our efforts to improve the quality of our publications.

1. **How did you use this manual? (Circle your response.)**

   (a) Installation (c) Maintenance (e) Training
   (b) Operation/use (d) Programming (f) Other (Please specify.) _____

2. **Did the manual meet your needs? Yes ☐ No ☐ Why?** _____

3. **Please rate the manual on the following categories. (Circle your response.)**

	Excellent	Good	Fair	Poor	Unacceptable
Accuracy	5	4	3	2	1
Clarity	5	4	3	2	1
Completeness	5	4	3	2	1
Table of Contents, Index	5	4	3	2	1
Illustrations, examples	5	4	3	2	1
Overall ease of use	5	4	3	2	1

4. **What things did you like *most* about this manual?** _____

5. **What things did you like *least* about this manual?** _____

6. **Please list and describe any errors you found in the manual.**

Page	Description/Location of Error
_____	_____
_____	_____
_____	_____
_____	_____

Name _____ Job Title _____
Street _____ Company _____
City _____ Department _____
State/Country _____ Telephone Number _____
Postal (ZIP) Code _____ Date _____

THANK YOU FOR YOUR COMMENTS AND SUGGESTIONS.

Please do not use this form to order manuals. Contact your representative at Digital Equipment Corporation or (in the USA) call our DECdirect℠ department at this toll-free number: 800-258-1710.

MYO

FOLD HERE AND TAPE. DO NOT STAPLE.

digital™

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO 33 MAYNARD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
Educational Services/Quality Assurance
12 Crosby Drive BUO/E08
Bedford, MA 01730-1493
USA

FOLD HERE AND TAPE. DO NOT STAPLE.