AlphaServer 4000/4100

Service Manual

Order Number: EK–4100A–SV. B01

This manual is for anyone who services an AlphaServer 4000/4100
pedestal or cabinet system. It includes troubleshooting information,
configuration rules, and instructions for removal and replacement of
field-replaceable units (FRUs).

Contents

Chapter 3 Troubleshooting

Chapter 4 Power System

Chapter 5 Error Logs

Chapter 6   Error Registers

Chapter 7   Removal and Replacement

Appendix A   Running Utilities

Appendix B   SRM Console Commands and Environment Variables

Appendix C   Operating the System Remotely

Index

Examples

Figures

Tables

Preface

Intended Audience

This manual is written for the customer service engineer.

Document Structure

This manual uses a structured documentation design. Topics are organized into small sections for efficient online and printed reference.   Each topic begins with an abastract, followed by an illustration or example, and ends with descriptive text.

This manual has seven chapters and three appendixes, as follows:

- **Chapter 1, System Overview,** introduces the DIGITAL AlphaServer 4000/4100 pedestal and cabinet systems and gives an overview of the system bus modules.

- **Chapter 2, Power-Up**, provides information on how to interpret the power-up display on the operator control panel, the console screen, and system LEDs. It also describes how hardware diagnostics execute when the system is initialized.

- **Chapter 3, Troubleshooting,** describes troubleshooting during power-up and booting, as well as the **test** command.

- **Chapter 4, Power System,** describes the AlphaServer 4000/4100 power system.

- **Chapter 5, Error Logs,** explains how to interpret error logs and how to use DECevent.

- **Chapter 6, Error Registers,** describes the error registers used to hold error information.

- **Chapter 7, Removal and Replacement,** describes removal and replacement procedures for field-replaceable units (FRUs).

- **Appendix A, Running Utilities**, explains how to run utilities such as the EISA Configuration Utility and RAID Standalone Configuration Utility.

- **Appendix B, SRM Console Commands and Environment Variables,** summarizes the commands used to examine and alter the system configuration.

- **Appendix C, Operating the System Remotely**, describes how to use the remote console monitor (RCM) to monitor and control the system remotely.

Documentation Titles

Table 1 lists titles related to AlphaServer 4000/4100 systems.

Table 1   AlphaServer 4000/4100 Documentation

Title	Order Number
ApphaServer 4100 User and Configuration Documentation Kit	QZ–00VAA–GZ
System Drawer User's Guide	EK–4100A–UG
Configuration and Installation Guide	EK–4100A–CG
ApphaServer 4000 User and Configuration Documentation Kit	QZ–00VAB–GZ
System Drawer User's Guide	EK–4000A–UG
Configuration and Installation Guide	EK–4100A–CG
Service Manual (hard copy)	EK–4100A–SV
Service Manual (diskette)	AK–QXBJB–CA
System Drawer Upgrades	EK–4041A–UI
PCI Upgrade	EK–4000A–UI
KN30n CPU Installation Card	EK–KN300–IN
MS3n0 Memory Installation Card	EK–MS300–IN
H7291 Power Supply Installation Card	EK–H7291–IN
ServerWORKS Manager Administrator User's Guide	ER–4QXAA–UA

Information on the Internet

Using a Web browser you can access the AlphaServer InfoCenter at:

http://www.digital.com/info/alphaserver/products.html

Access the latest system firmware either with a Web browser or via FTP as follows:

ftp://ftp.digital.com/pub/Digital/Alpha/firmware/

Interim firmware released since the last firmware CD is located at:

ftp://ftp.digital.com/pub/Digital/Alpha/firmware/interim/

Chapter 1

System Overview

This chapter introduces the DIGITAL AlphaServer 4000 and the DIGITAL AlphaServer 4100 systems. These systems are available in cabinets or pedestals.

There are three system drawers; two, the BA30B and the BA30C, are used in the AlphaServer 4000, and the third, the BA30A, is used in the AlphaServer 4100.

The pedestal system has one system drawer and up to three StorageWorks shelves. The cabinet system can have a combination of system drawers and StorageWorks shelves that occupy the five sections of the cabinet.

Topics in this chapter include the following:

- AlphaServer 4100 System Drawer (BA30A)
- AlphaServer 4000 System Drawer (BA30C)
- AlphaServer 4000 System Drawer (BA30B)
- Cabinet System
- Pedestal System
- Control Panel and Drives
- System Consoles
- System Architecture
- System Motherboard
- CPU Types
- Memory Modules
- Memory Addressing
- System Bus
- System Bus to PCI Bus Bridge Module
- PCI I/O Subsystem
- Server Control Module
- Power Control Module
- Power Supply

1.1 AlphaServer 4100 System Drawer (BA30A)

Components in the BA30A system drawer are located in the system bus card cage, the PCI card cage, the control panel assembly, and the power and cooling section. The drawer measures 30 cm x 45 cm (11.8 in. x 17.7 in.) and fully configured weighs approximately 45.5 kg (~100 lbs).

Figure 1-1   Components of the BA30A System Drawer
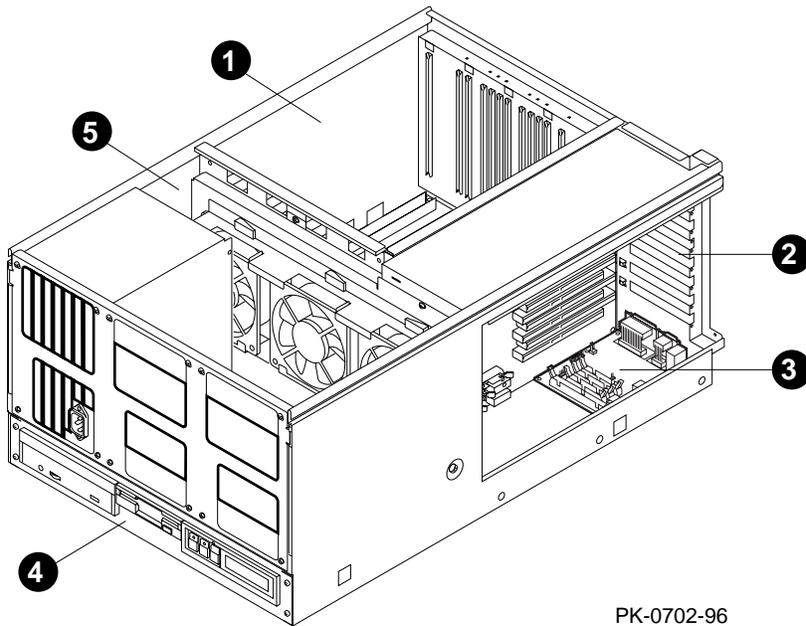
PK-0702-96

When the system drawer is in a pedestal, the control panel assembly is mounted in a tray at the top of the drawer.

The numbered callouts in Figure 1-1 refer to components of the system drawer.

❶ System card cage, which holds the system motherboard and the CPU, memory, bridge, and power control modules. (The difference between the BA30A and the BA30C is the system motherboard.)

❷ PCI/EISA card cage, which holds the PCI motherboard, option cards, and server control module.

❸ Server control module, which holds the I/O connectors and remote console monitor.

❹ Control panel assembly, which includes the control panel, a floppy drive, and a CD-ROM drive.

❺ Power and cooling section, which contains one to three power supplies and fans.

Cover Interlocks

The system drawer has three cover interlocks: one for the system bus card cage, one for the PCI card cage, and one for the power and system fan area.

Figure 1-2 Cover Interlock Circuit
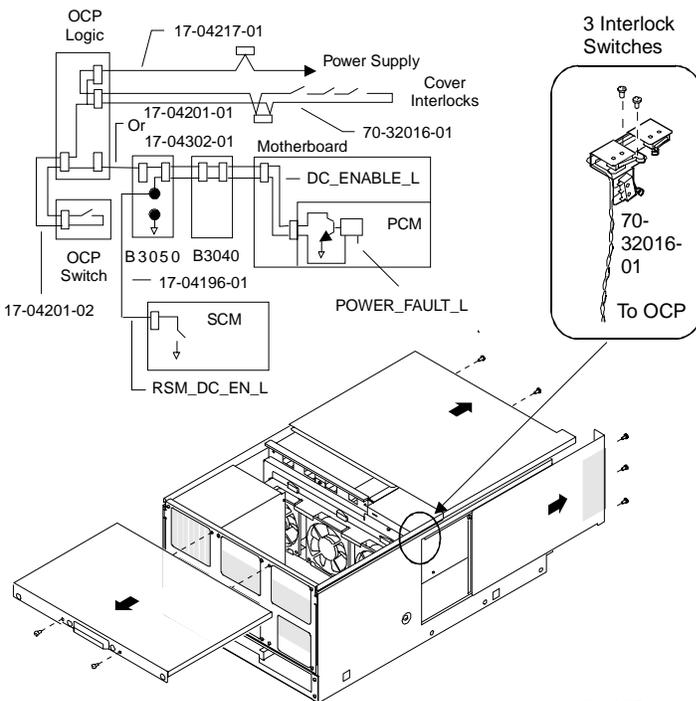
PKW-0403-95

NOTE: The cover interlocks must be engaged to enable power-up.

To override the cover interlocks, find a suitable object to close the interlock circuit.

1.2 AlphaServer 4000 System Drawer (BA30C)

Components in the BA30C system drawer are located in the system bus card cage, PCI card cage, control panel assembly, and power and cooling section. The drawer measures 30 cm x 45 cm (11.8 in. x 17.7 in.) and fully configured weighs approximately 45.5 kg (~100 lbs).

Figure 1-3   Components of the BA30C System Drawer
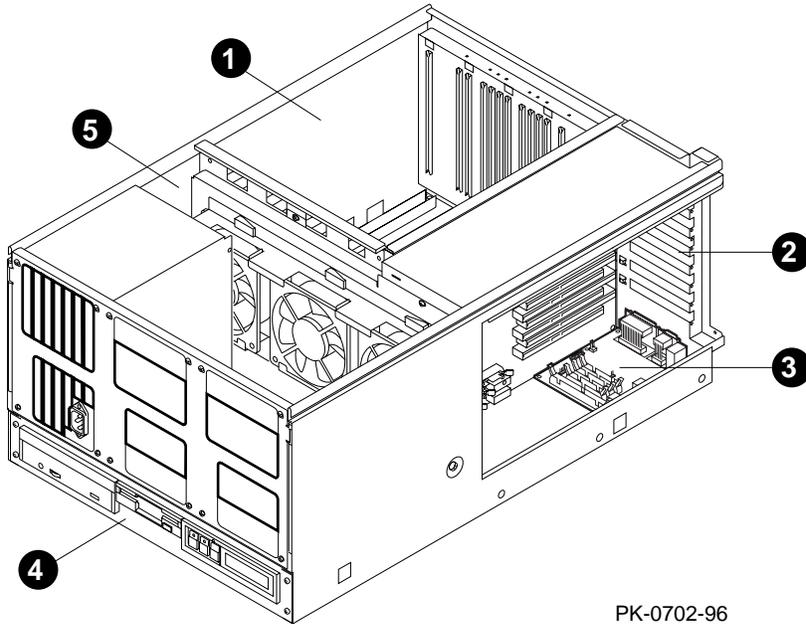
PK-0702-96

When the system drawer is in a pedestal, the control panel assembly is mounted in a tray at the top of the drawer.

The numbered callouts in Figure 1-3 refer to components of the system drawer.

❶ System card cage, which holds the system motherboard and the CPU, memory, bridge, and power control modules. (The difference between the BA30A and the BA30C is the system motherboard.)

❷ PCI/EISA card cage, which holds the PCI motherboard, option cards, and server control module.

❸ Server control module, which holds the I/O connectors and remote console monitor.

❹ Control panel assembly, which includes the control panel, a floppy drive, and a CD-ROM drive.

❺ Power and cooling section, which contains one to three power supplies and fans.

Cover Interlocks

The system drawer has three cover interlocks: one for the system bus card cage, one for the PCI card cage, and one for the power and system fan area.

Figure 1-4   Cover Interlock Circuit
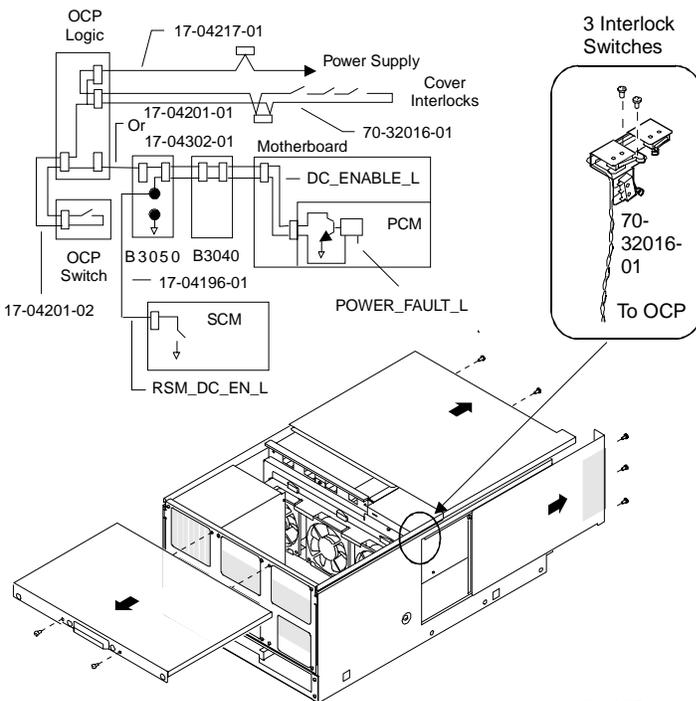
PKW-0403-95

NOTE: The cover interlocks must be engaged to enable power-up.

To override the cover interlocks, find a suitable object to close the interlock circuit.

1.3 AlphaServer 4000 System Drawer (BA30B)

Components in the BA30B system drawer are located in the system bus card cage, *two* PCI card cages, the control panel assembly, and the power and cooling section. The drawer measures 30 cm x 45 cm (11.8 in. x 17.7 in.) and fully configured weighs approximately 45.5 kg (~100 lbs).

Figure 1-5   Components of the BA30B System Drawer
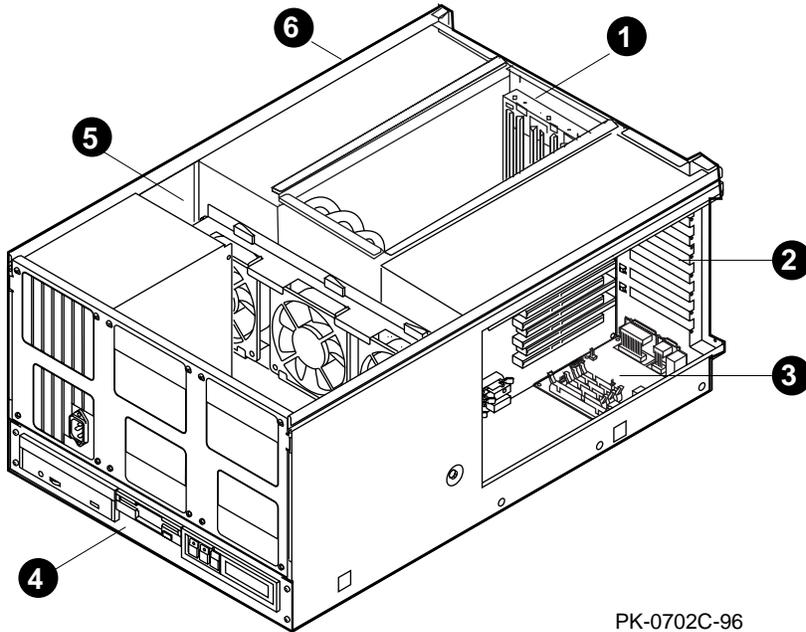
PK-0702C-96

When the system drawer is in a pedestal, the control panel assembly is mounted in a tray at the top of the drawer.

The numbered callouts in Figure 1-5 refer to components of the system drawer.

❶ System card cage holds the system motherboard, the CPU, memory, bridge, and power control modules.

❷ PCI/EISA card cage holds the PCI/EISA motherboard for PCI/EISA 0 and PCI 1, option cards, and server control module.

❸ Server control module holds the I/O connectors and remote console.

❹ Control panel assembly holds the control panel, a floppy, and a CD-ROM.

❺ Power and cooling section contains one to three power supplies and three fans.

❻ PCI card cage holds the PCI motherboard for PCI 2 and PCI 3.

Cover Interlocks

The system drawer has four cover interlocks: one for each section of the drawer.

Figure 1-6   Cover Interlock Circuit
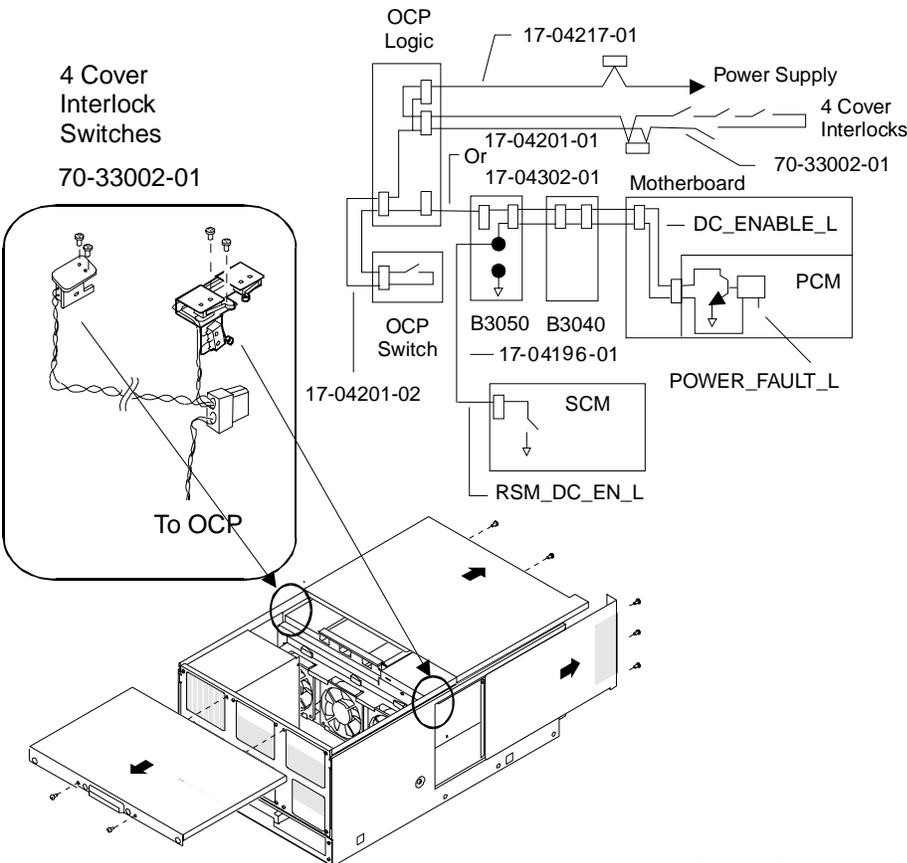
PKW-0403K-96

1.4 Cabinet System

The AlphaServer 4000/4100 cabinet system can accommodate multiple systems in a single cabinet.   There are four cabinet variations that can hold different system configurations.   Diferences are in power distribution and drawer mounting; from the outside the cabinets look almost identical.

Figure 1-7   AlphaServer 4000/4100 Cabinet System
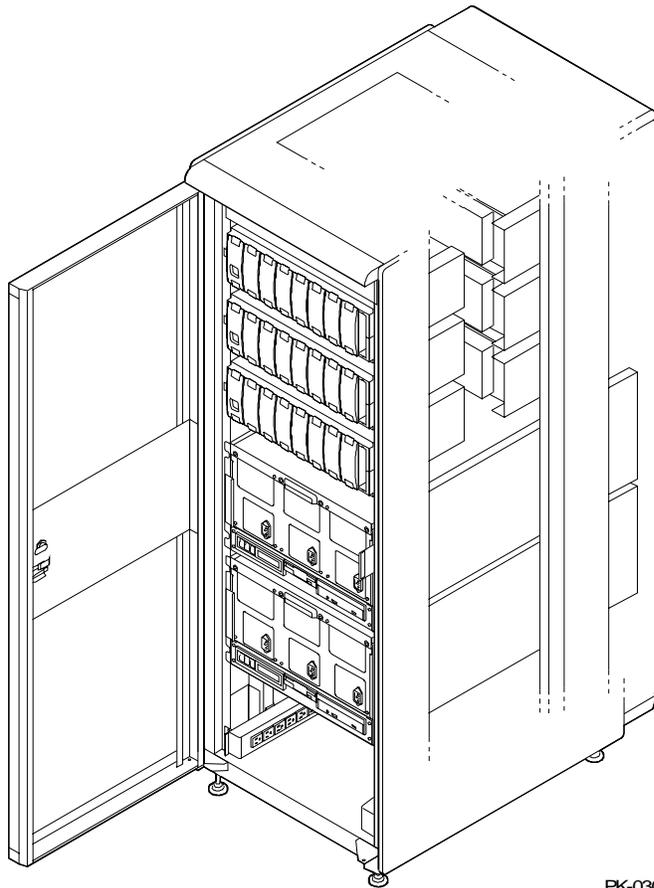
PK-0306-96

Cabinet Differences

Cabinet	Power	Mounting	Destination
H9A10-EB	AC input box power strips	C channel (max drawers: 4)	North America Asia Pacific
H9A10-EC	AC input box power strips	C channel (max drawers: 4)	Europe
H9A10-EL	Two 120 volt H7600-AA power controllers	Pull-out tray (max drawers: 3)	North America Asia Pacific
H9A10-EM	Two 240 volt H7600-DB power controllers	Pull-out tray (max drawers: 3)	Europe

Cabinet System Fan Tray

At the top of cabinet systems is a fan tray containing three exhaust fans, a small 12-volt power supply, and a module that distributes power to the server control module in each drawer.

Figure 1-8   Cabinet Fan Tray

PKW0441A-96
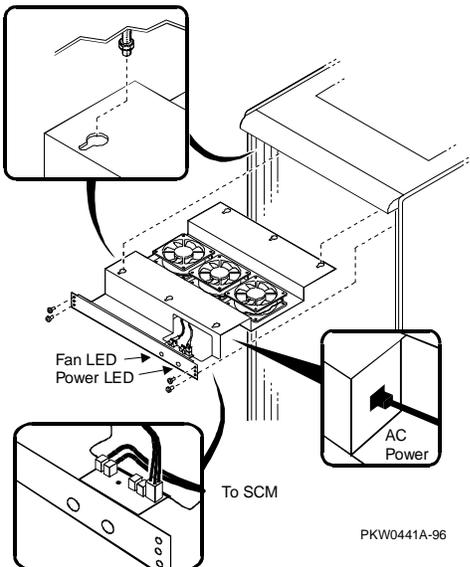
1.5 Pedestal System

The pedestal system contains one system drawer with a control panel, a CD-ROM drive, and a floppy drive. In the pedestal control panel area there is space for an optional tape or disk drive. Three StorageWorks shelves provide up to 90 Gbytes of in-cabinet storage.

Figure 1-9   Pedestal System Front
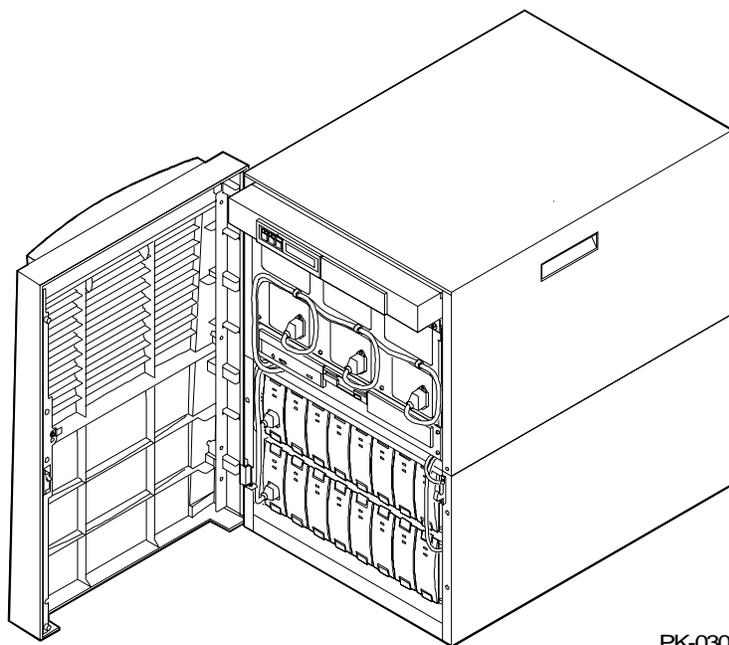
PK-0301-96

In the pedestal system, the control panel is located at the top left in a tray. See Figure 1-11. There is space for an optional device beside it.

Figure 1-10   Pedestal System Rear
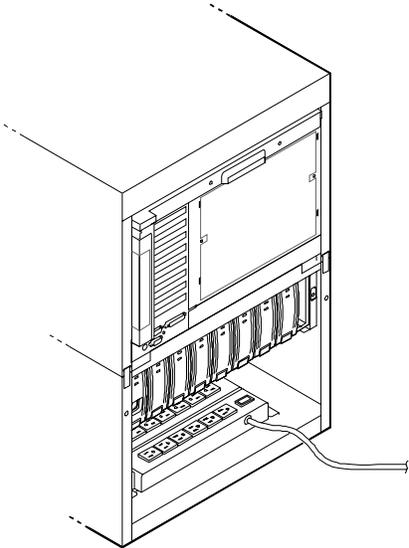
PK-0307a-96

1.6 Control Panel and Drives

The control panel includes the On/Off, Halt, and Reset buttons and a display. In a pedestal system the control panel is located in a tray at the top of the system drawer. In a cabinet system it is at the bottom of the system drawer with the CD-ROM drive and the floppy drive.

Figure 1-11   Control Panel Assembly

PK-0751-96
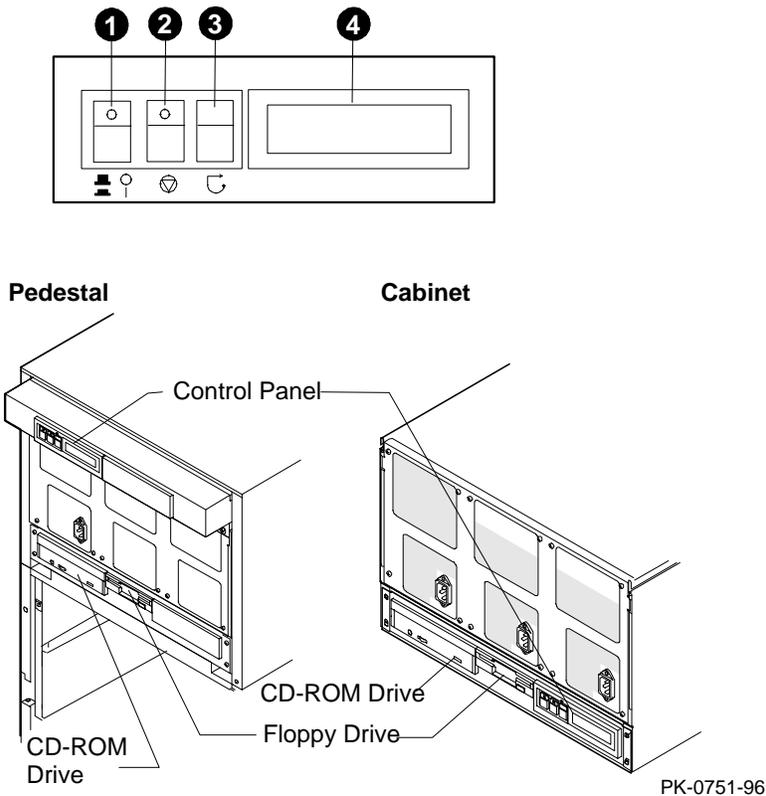
❶   On/Off button. Powers the system drawer on or off. When the LED at the top of the button is lit, the power is on.   The On/Off button is connected to the power supplies and the system interlocks.

NOTE: The LEDs on some modules are on when the line cord is plugged in, regardless of the position of the On/Off button.

❷   Halt button. Pressing this button in (so the LED at the top of the button is on) does the following:

If DIGITAL UNIX or OpenVMS is running, halts the operating system and returns to the SRM console. The Halt button has no effect on Windows NT.

If the Halt button is in when the system is reset or powered up, the system halts in the SRM console, regardless of the operating system. DIGITAL UNIX and OpenVMS systems that are configured for autoboot will not boot if the Halt button is in. Windows NT systems halt in the SRM console; AlphaBIOS is not loaded and started.

If you press the Halt button in (LED on) and do not issue commands that disturb the system state, entering the **continue** command returns the system to the operating system it was running. To return to console mode again, press the Halt button (LED off) and then press it again (LED on).

If the system is hung, pressing the Halt button (LED on) usually brings up the SRM console. Enter the **crash** command to do a crash dump. If pressing the Halt button does not bring up the SRM console, there is probably a hardware fault that is not allowing the halt signal to pass from the XBUS to the CPU.

❸   Reset button. Initializes the system drawer. If the Halt button is pressed (LED on) when the system is reset, the SRM console is loaded and remains in the system regardless of any other conditions.

❹   Control panel display. Indicates status during power-up and self-test. The OCP display is a 16-character LCD.   Its controller is on the XBUS on the PCI motherboard.

While the operating system is running, displays the system type as a default. This message can be changed by the user.

CD-ROM drive. The CD-ROM drive is used to load software, firmware, and updates. Its controller is on PCI1 on the PCI motherboard.

Floppy disk drive. The floppy drive is used to load software and firmware updates. The floppy controller is on the XBUS on the PCI motherboard.

1.7 System Consoles

There are two console programs: the SRM console and the AlphaBIOS console.

SRM Console Prompt

On systems running the DIGITAL UNIX or OpenVMS operating system, the following console prompt is displayed after system startup messages are displayed, or whenever the SRM console is invoked:

```
P00>>>
```

NOTE: The console prompt displays only after the entire power-up sequence is complete. This can take up to several minutes if the memory is very large.

AlphaBIOS Boot Menu

On systems running the Windows NT operating system, the Boot menu is displayed when the AlphaBIOS console is invoked:
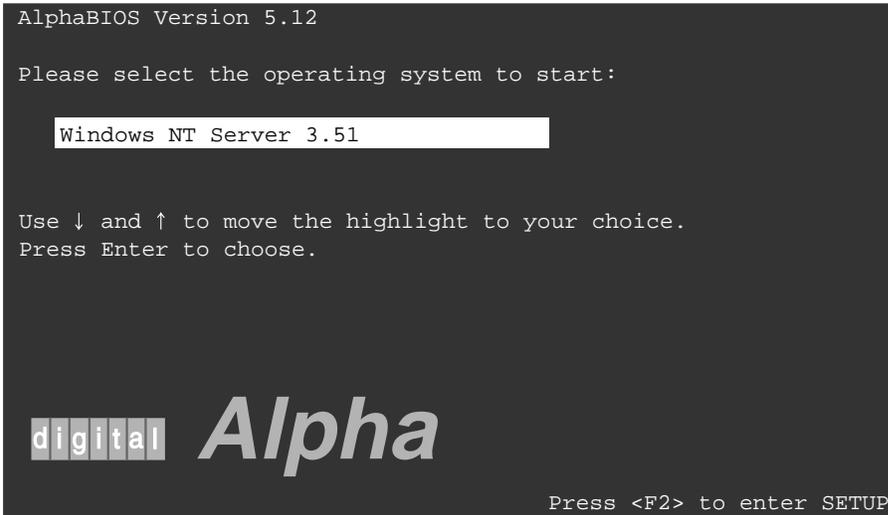
```
AlphaBIOS Version 5.12

Please select the operating system to start:

   Windows NT Server 3.51


Use ↓ and ↑ to move the highlight to your choice.
Press Enter to choose.




digital  Alpha

                                   Press <F2> to enter SETUP
```

PK-0728-96

SRM Console

The SRM console is a command-line interface that is used to boot the DIGITAL UNIX and OpenVMS operating systems. It also provides support for examining and modifying the system state and configuring and testing the system. The SRM console can be run from a serial terminal or a graphics monitor.

AlphaBIOS Console

The AlphaBIOS console is a menu-based interface that supports the Microsoft Windows NT operating system. AlphaBIOS is used to set up operating system selections, boot Windows NT, and display information about the system configuration. The EISA Configuration Utility and the RAID Standalone Configuration Utility are run from the AlphaBIOS console. AlphaBIOS runs on either a serial or graphics terminal, but Windows NT requires a graphics monitor.

Environment Variables

Environment variables are software parameters that define, among other things, the system configuration. They are used to pass information to different pieces of software running in the system at various times. The **os_type** environment variable, which can be set to VMS, UNIX, or NT, determines which of the two consoles is to be used. The SRM console is always brought into memory, but AlphaBIOS is loaded if **os_type** is set to **NT** and the Halt button is out (not lit).

Refer to Appendix B of this guide for a list of the environment variables used to configure AlphaServer 4000 and 4100 systems.

Refer to the *AlphaServer 4x00 System Drawer User's Guide* for information on setting environment variables.

It is recommended that you keep a record of the environment variables for each system that you service. Some environment variable settings are lost when a module is swapped and must be restored after the new module is installed. Refer to Appendix B for a convenient worksheet for recording environment variable settings.

1.8 System Architecture

**Alpha microprocessor chips are used in these systems. The CPU, memory, and
the I/O bridge module(s) are connected to the system bus motherboard.**

Figure 1-12   Architecture Diagram
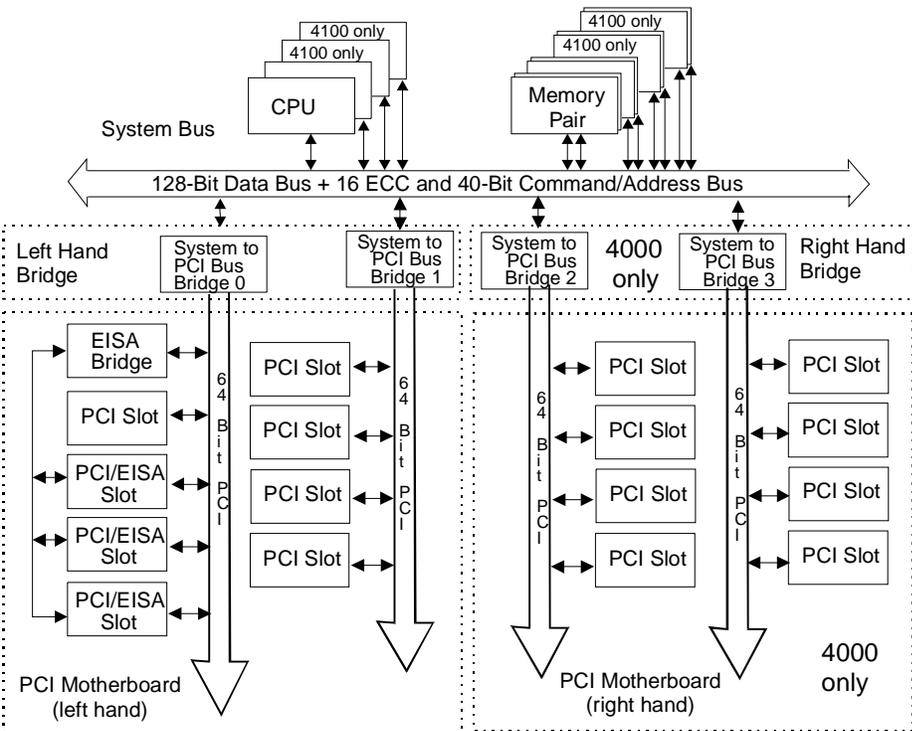
PKW0421E-96

AlphaServer 4000/4100 systems use the Alpha chip for the CPU. The CPU, memory, and I/O bridge modules, one to PCI/EISA I/O buses and another (4000 only) to another pair of PCIs, are connected to the system bus motherboard. A fourth type of module, the power control module, also plugs into the system motherboard. A fully configured 4100 system drawer can have up to four CPUs, four memory pairs, and a total of eight I/O options. The I/O options can be all PCI options or a combination of PCI options and EISA options, but there can be no more than three EISA options. A fully configured 4000 system drawer can have up to two CPUs, two memory pairs, and a total of sixteen I/O options. The I/O options can be all PCI options or a combination of PCI options and EISA options, but there can be no more than three EISA options.

The system bus has a 144-bit data bus protected by 16 bits of ECC and a 40-bit command/address bus protected by parity. The bus speed depends on the speed of the CPU in slot 0 which provides the clock for the buses. The 40-bit address bus can create one terabyte of addresses (that's a million billion). The bus connects CPUs, memory, and the system bus to PCI bus bridge(s).

The CPU modules are available with and without an external cache. The Alpha chip has an 8-Kbyte instruction cache (I-cache), an 8-Kbyte write-through data cache (D-cache), and a 96-Kbyte, write-back secondary data cache (S-cache). Some variants of the CPU module include an onboard cache. The cache system is write-back. The system drawer supports up to four CPUs.

The memory modules are placed on the system motherboard in pairs. Each module drives half of the system bus, along with the associated ECC bits. Memory pairs consist of two modules that are the same size and type. Two types are available: synchronous and asynchronous (EDO) memory.

The system bus to PCI bus bridge module translates system bus commands and data addressed to I/O space to PCI commands and data. It also translates PCI bus commands and data addressed to system memory or CPUs to system bus commands and data. The PCI bus is a 64-bit wide bus used for I/O. Both the 4100 and the 4000 have one PCI/EISA card cage, and the 4000 may contain a second PCI card cage.

The power control module, which is on the system motherboard, monitors power and the system environment.

1.9 System Motherboard

The system motherboard is on the floor of the system card cage. It has slots for the CPU, memory, power control, and bridge modules.

Figure 1-13   System Motherboard Module Locations

4100 Motherboard (54-23803-01)
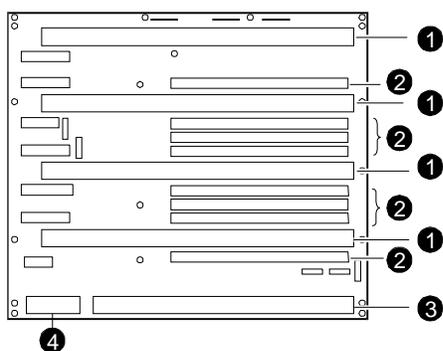
4000 Motherboard (54-23803-02)
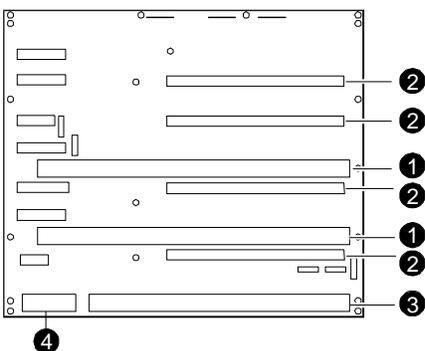
4000 Motherboard (54-23805-01)
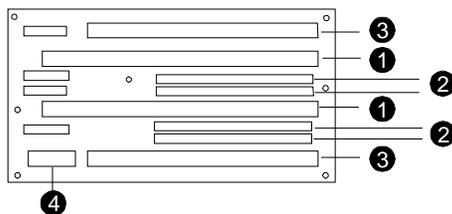
PKW0440J-96

The system motherboard has the logic for the system bus. It is the backplane that holds the CPU, memory, bridge, and power control modules. Figure 1-13 shows diagrams of the three motherboards used in AlphaServer 4000/4100 systems. The module locations are designated by the callouts.

❶   CPU module

❷   Memory module

❸   Bridge module

❹   Power control module

1.10 CPU Types

AlphaServer 4000 and 4100 systems can be configured with one of several CPU variants. Variants are differentiated by CPU speeds and the presence or absence of a backup data cache external to the Alpha microprocessor chip.

Figure 1-14   CPU Module Layout

Typical Uncached CPU

Typical Cached CPU
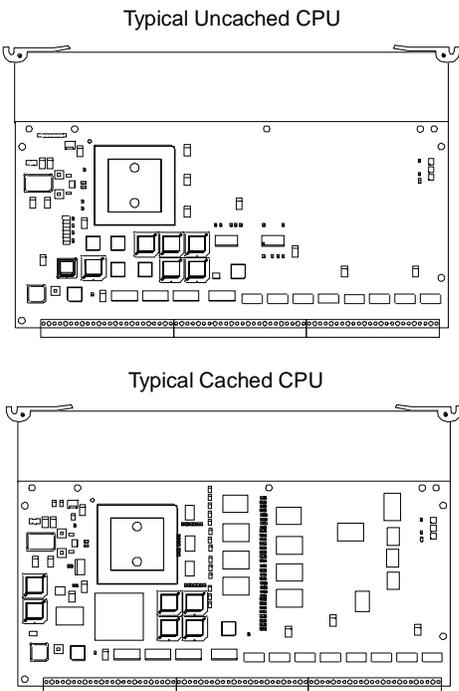
PKW0422C-96

Alpha Chip Composition

The Alpha chip is made using state-of-the-art chip technology, has a transistor count of 9.3 million, consumes 50 watts of power, and is air cooled (a fan is on the chip). The default cache system is write-back and when the module has an external cache, it is write-back.

Chip Description

Unit	Description
Instruction	8-byte cache, 4-way issue
Execution	4-way execution; 2 integer units, 1 floating-point adder, 1 floating-point multiplier
Memory	Merge logic, 8-Kbyte write-through first-level data cache, 96-Kbyte write-back second-level data cache, bus interface unit

CPU Variants

Module Variant	Clock Frequency	Onboard Cache
B3001-CA	300 MHz	None
B3002-AB	300 MHz	2 Mbytes
B3004-BA	300 MHz	2 Mbytes
B3004-AA	400 MHz	4 Mbytes
B3004-DA	466 MHz	4 Mbytes

CPU Configuration Rules

- The first CPU must be in CPU slot 0 to provide the system clock.

- Additional CPU modules should be installed in ascending order by slot number.

- All CPUs must have the same Alpha chip clock speed. The system bus will hang without an error message if the oscillators clocking the CPUs are different.

- Mixing of cached and uncached CPUs is not supported.

Color Codes

The top edge of the CPU module variant is color coded for easy identification.

Color	Option Number	Description
Dark Blue	B3001-CA	300 MHz, uncached
Green	B3002-AB	300 MHz, 2MB cached
Green	B3004-BA	300 MHz, 2MB cached
Orange	B3004-AA	400 MHz, 4MB cached
Red	B3004-DA	466 MHz, 4MB cached

1.11 Memory Modules

Memory modules are used only in pairs — two modules of the same size and type. Each module provides either the low half or the high half of the memory space. The 4100 system drawer can hold up to four memory module pairs. The 4000 system drawer can hold up to two memory module pairs.

Figure 1-15   Memory Module Layout

Typical Synchronous Memory

Typical EDO Memory
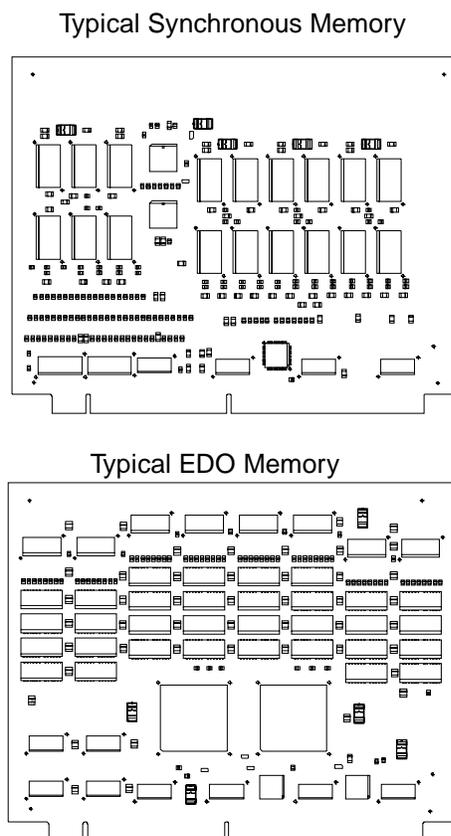
PKW0423C-96

Memory Variants

Each memory option consists of two identical modules. Each 4100 drawer supports up to four memory options, for a total of 4 Gbytes of memory: 4000 drawers support half that. Memory modules are used only in pairs and are available in 128 Mbyte, 512 Mbyte, and 1 Gbyte sizes. The 128-Mbyte option is synchronous memory, while the larger sizes are asynchronous memory (EDO).

			DRAM		
Option	Size	Module	Type	Number	Size
MS320-CA	128 MB	B3020-CA	Synch.	36	4 MB x 4
MS330-EA	512 MB	B3030-EA	Asynch. (EDO)	144	4 MB x 4
MS330-FA	1 GB	B3030-FA	Asynch. (EDO)	72	16 MB x 4
MS330-GA	2 GB	B3030-GA	Asynch. (EDO)	144	16 MB x 4

Memory Operation

Memory modules are used only in pairs; each module provides half the data, or 64 bits plus 8 ECC bits, of the octaword (16 byte) transferred on the system bus. Modules are placed in slots designated MEMxL and MEMxH.

NOTE: Modules in slots MEMxL do not drive the lower 8 bytes, and modules in slots MEMxH do not drive the higher 8 bytes of the 16 byte transfer.

Unless otherwise programmed, memory drives the system bus in bursts. Upon each memory fetch, data is transferred in 4 consecutive cycles transferring 64 bytes. There are situations, however, when memories made with EDO DRAMs cannot provide data fast enough to complete the system bus transactions. When these situations arise, EDO type memories assert a signal that causes the system bus to stall for one (occasionally more) clock tick. When memory completes such an operation, it releases the system bus.

Memory Configuration Rules

In a system, memories of different sizes and types are permitted, but:

- Memory modules are installed and used in pairs. Both modules in a memory pair must be of the same size and type.

- The largest memory pair must be in slots MEM 0L and MEM 0H.

- Other memory pairs must be the same size or smaller than the first memory pair.

- Memory pairs must be installed in consecutive slots.

1.12 Memory Addressing

Alpha system memory addressing is unusual because memory address space is determined not by the amount of physical memory but is calculated by a multiple of the size of the memory pair in slot MEM0x.

Figure 1-16   How Memory Addressing Is Calculated

Fourth pair address space
512 Mbyte space empty

Third pair address space
512 Mbyte 1/2 occupied
(2 B3020-DA - 128 Mbyte/mod)

Second pair address space
512 Mbyte 1/2 occupied
(2 B3020-DA - 128 Mbyte/mod)

First pair defines
total address space
always fully occupied
(2 B3020-EA 256 Mbyte/mod)

2028 Mbyte

1536 Mbyte

1024 Mbyte

Address hole

512 Mbyte

0
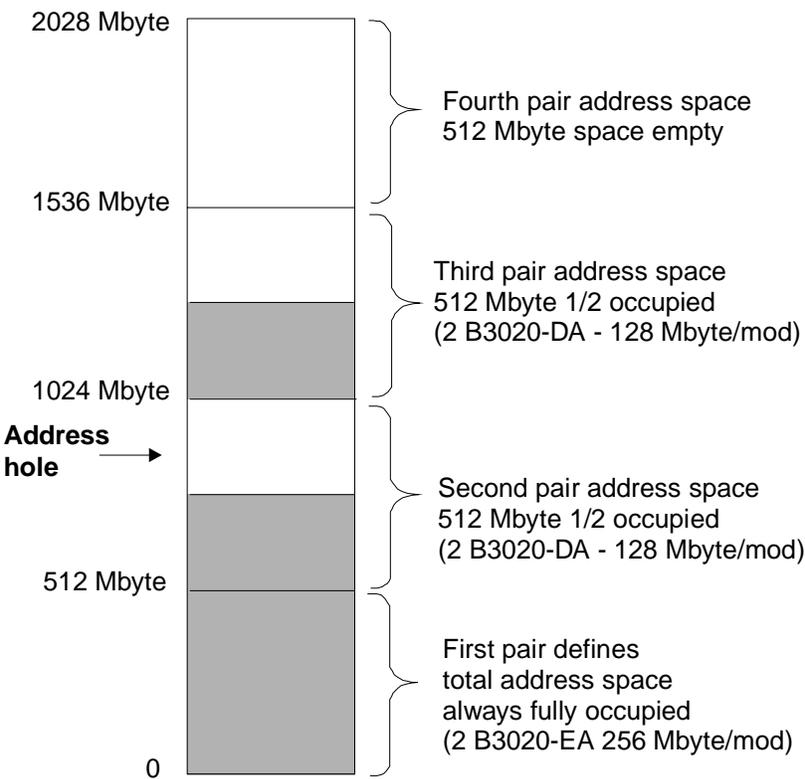
PKW0424-96

The rules for addressing memory are as follows:

1. Address space is determined by the memory pair in slot MEM0.

2. Memory pairs need not be the same size.

3. The memory pair in slot MEM0 must be the largest of all memory pairs. Other memory pairs may be as large but none may be larger.

4. The starting address of each memory pair is N times the size of the memory pair in slot MEM0. N=0,1,2,3.

5. Memory addresses are contiguous within each module pair.

6. If memory pairs are of different sizes, memory "holes" can occur in the physical address space. See Figure 1-16.

7. Software creates contiguous virtual memory even though physical memory may not be contiguous.

1.13 System Bus

The system bus consists of a 40-bit command/address bus, a 128-bit plus ECC data bus, and several control signals and clocks.

Figure 1-17   System Bus Block Diagram
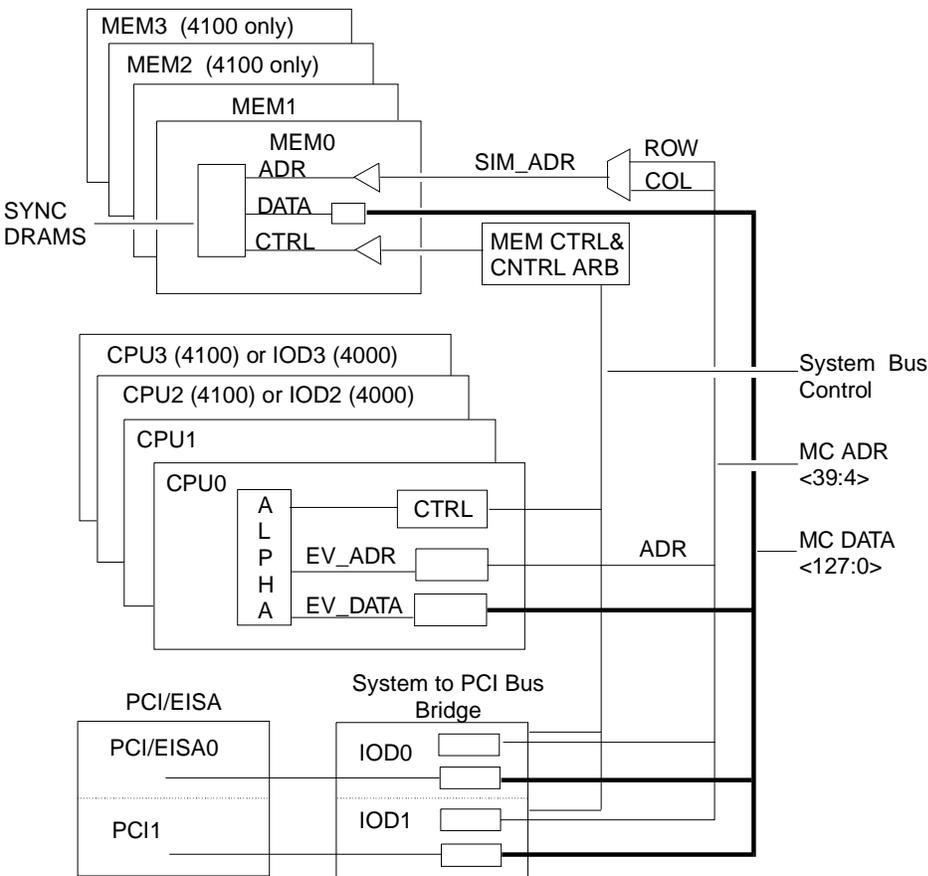
PKW0425-96

The system bus motherboard consists of a 40-bit command/address bus, a 128-bit plus ECC data bus, and several control signals, clocks, and a bus arbiter. The bus requires that all CPUs have the same high-speed oscillator providing the clock to the Alpha chip.

The AlphaServer 4100 system bus connects up to four CPUs, four pairs of memory modules, and a single I/O bus bridge module. Note that the I/O bus bridges may be desinated as IODn where n is the number of the PCI bus. The first bridge is designated IOD0 and IOD1.

The AlphaServer 4000 system bus connects up to two CPUs, two pairs of memory modules, and two I/O bus bridge modules. The second bridge on the 4000 system bus is designated IOD2 and IOD3.

The system bus clock is provided by an oscillator on the CPU in slot CPU0. This oscillator has a 1:5 ratio to the Alpha chip. With 300 MHz CPUs, for example, the system bus operates at 60 MHz.

The system bus motherboard initiates memory refresh transactions. The motherboard sits at the bottom of the system drawer, and in addition to CPUs, memory, and I/O bridges, holds a power control module.

5 volt and 3.43 volt power is provided directly to the motherboard from the power supplies.

1.14 System Bus to PCI Bus Bridge Module

The bridge module is the physical interconnect between the system motherboard and any PCI motherboard in the system.

Figure 1-18   Bridge Module
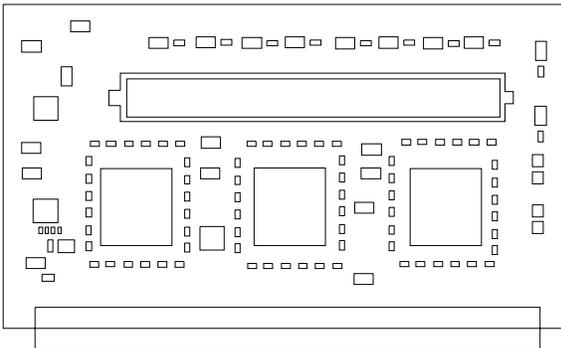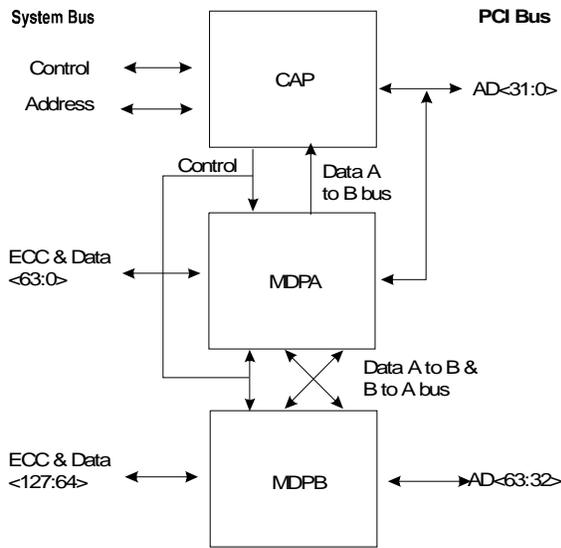
PKW0426r-96

The system bus to PCI bus bridge module converts system bus commands and data addressed to I/O space to PCI commands and data; and converts PCI bus commands and data addressed to system memory or CPUs to system bus commands and data. An AlphaServer 4100 system has one bridge module; an AlphaServer 4000 system can have a second bridge module.

The bridge has two major components:

- Command/address processor (CAP) chip

- Two data path chips (MDPA and MDPB)

There are two sets of these three chips, one set on each side of the module. Each set bridges to one of the PCI buses on the PCI motherboard.

The interface on the system bus side of the bridge responds to system bus commands addressed to the upper 64 Gbytes of I/O space. I/O space is addressed whenever bit <39> on the system bus address lines is set. The space so defined is 512 Gbytes in size. The first 448 Gbytes are reserved and the last 64 Gbytes, when bits <38:36> are set, are mapped to the PCI I/O buses.

The interface on the PCI side of the bridge responds to commands addressed to CPUs and memory on the system bus. On the PCI side, the bridge provides the interface to the PCIs. Each PCI bus is addressed separately. The bridge does not respond to devices communicating with each other on the same PCI bus. However, should a device on one PCI address a device on the other PCI bus, commands, addresses, and data run through the bridge out onto the system bus and back through the bridge to the other PCI bus.

In addition to its bridge function, the system bus to PCI bus bridge module monitors every transaction on the system bus for errors. It monitors the data lines for ECC errors and the command/address lines for parity errors.

NOTE: When errors are logged, the two bridge modules on the AlphaServer 4000 are differentiated in the error log by their engineering code names, the left hand horse and the right hand horse. The left hand horse is the B3040-AA module; it is in the left most slot on the system bus motherboard when seen from the rear of the drawer. The right hand horse is the B3040-AB module, and it is in the right most slot on the system bus motherboard when seen from the rear of the drawer.

1.15 PCI I/O Subsystem

The I/O subsystem is PCI. Both the 4100 and the 4000 have two four-slot PCI buses that hold up to eight I/O options. One of these buses can be both PCI and EISA but can hold not more than four options three of which may be EISA. The 4000 can have an additional two four-slot PCI buses allowing a total of sixteen I/O options.

Figure 1-19 PCI Block Diagram
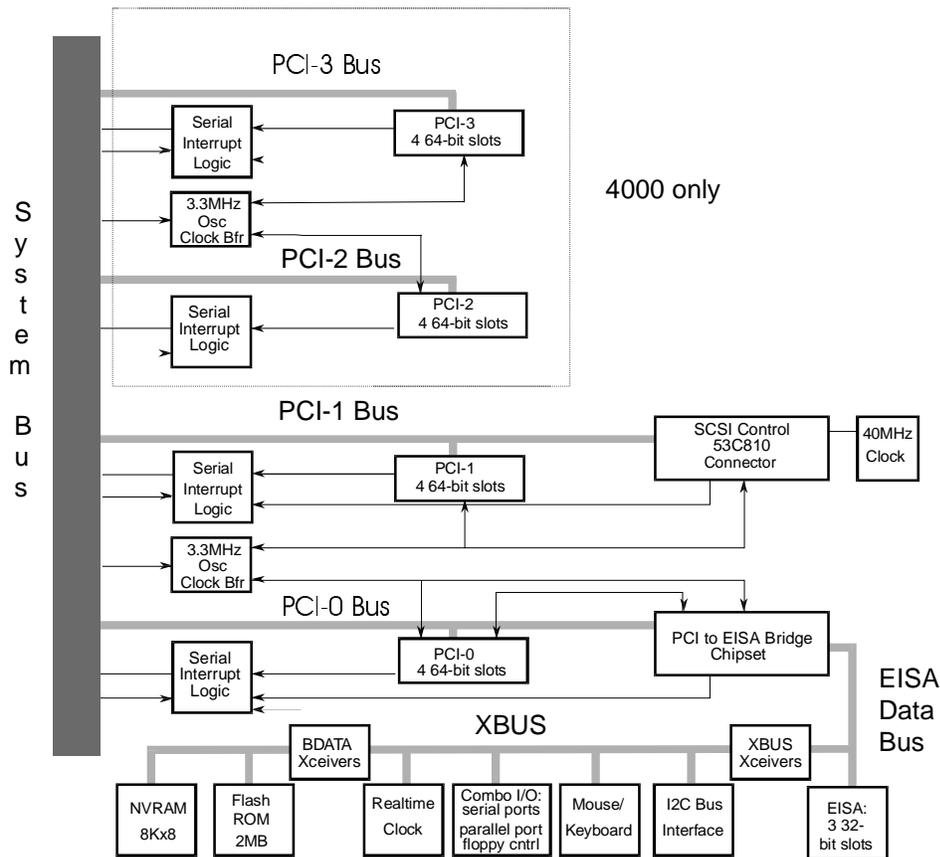
PKW0431R-96

Table 1–1   PCI Motherboard Slot Numbering

Slot	PCI0	PCI1	PCI2 (4000 only)	PCI3 (4000 only)
0	Reserved	Reserved	Reserved	Reserved
1	PCI to EISA bridge	Internal CD-ROM controller	Reserved	Reserved
2	PCI or EISA slot	PCI slot	PCI slot	PCI slot
3	PCI or EISA slot	PCI slot	PCI slot	PCI slot
4	PCI or EISA slot	PCI slot	PCI slot	PCI slot
5	PCI slot	PCI slot	PCI slot	PCI slot

The logic for two PCI buses is on each PCI motherboard.

- PCI0 is a 64-bit bus with a built-in PCI to EISA bus bridge. PCI0 has one dedicated PCI slot and three slots, though there are six connectors, that can be PCI or EISA slots. Each slot has an EISA connector and a PCI connector only one of which may be used at a time. PCI0 is powered by 5V.

- PCI1 is a 64-bit bus with a built-in CD-ROM controller and four PCI slots. PCI1 is powered by 5V.

- PCI2 (4000 only) is a 64-bit four-slot PCI bus powered by both 3V and 5V.

- PCI3 (4000 only) is a 64-bit four-slot PCI bus powered by both 3V and 5V.

The B3050-AA PCI motherboard has cable connections to remote I/O (mouse, keyboard, serial port, and parallel port), an internal floppy drive, an internal CD-ROM drive, the control panel, and 5V power. Also on this module are the chips for the PCI to EISA bridge and the internal CD-ROM controller. This module is the motherboard for the PCI card cage on the left side of the system drawer.

An 8-bit XBUS is connected to the EISA bus. On this bus there is an interface to the system I^{2}C bus; mouse and keyboard support; an I/O combo controller supporting two serial ports, the floppy controller, and a parallel port; a real-time clock; two 1-Mbyte flash ROMs containing system firmware, and an 8-Kbyte NVRAM.

The B3050-AB PCI motherboard, used only in the AlphaServer 4000, contains two four-slot 64-bit PCI buses.

1.16 Server Control Module

The server control module enables remote console connections to the system drawer. The module passes signals to COM ports 1 and 2, the keyboard, and the mouse to the standard I/O connectors.

Figure 1-20 Server Control Module

Standard I/O

Remote Console
Monitor

PK-0702B-96
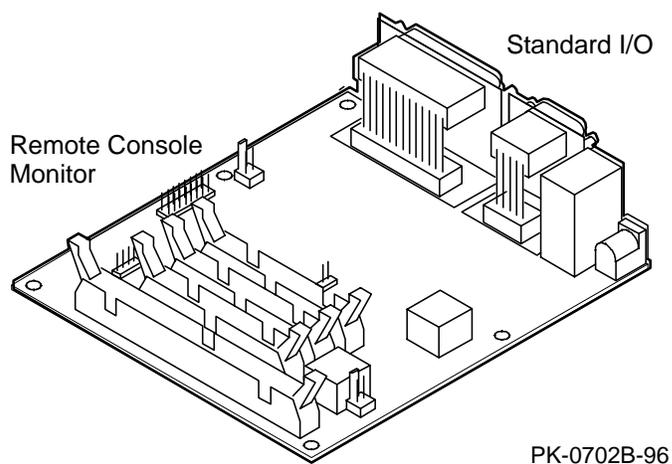
The server control module has two sections: the remote console monitor (RCM) and the standard I/O. See Appendix C for information on controlling the system remotely.

The remote console monitor connects to a modem through the modem port on the bulkhead. The RCM requires a 12V power connection.

The standard I/O ports (keyboard, mouse, COM1 and COM2 serial, and parallel ports) are on the same bulkhead.

1.17 Power Control Module

The power control module controls power sequencing and monitors power supply voltage, temperature, and fans.

Figure 1-21   Power Control Module

System Motherboard
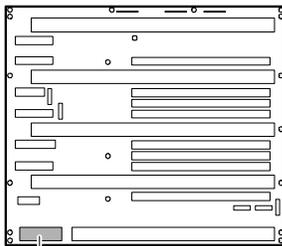
Power Control
Module Slot
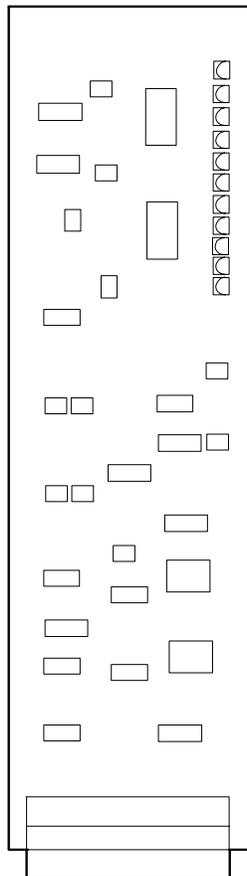
PK-0710-96

The power control module performs these functions:

- Controls power sequencing.

- Monitors the combined output of power supplies and shuts down power if it is not in range.

- Monitors system temperature and shuts off power if it is out of range.

- Monitors the fans in the system drawer and on the CPU modules and shuts down power if a fan fails.

- Provides visual indication of faults through LEDs.

1.18 Power Supply

The system drawer power supplies provide power only to components in the drawer. One or two power supplies are required, depending on the number of CPU modules and PCI card cages; a second or third can be added for redundancy. The power system is described in detail in Chapter 4.

Figure 1-22   Location of Power Supply

Power Supply 2
Power Supply 1
Power Supply 0
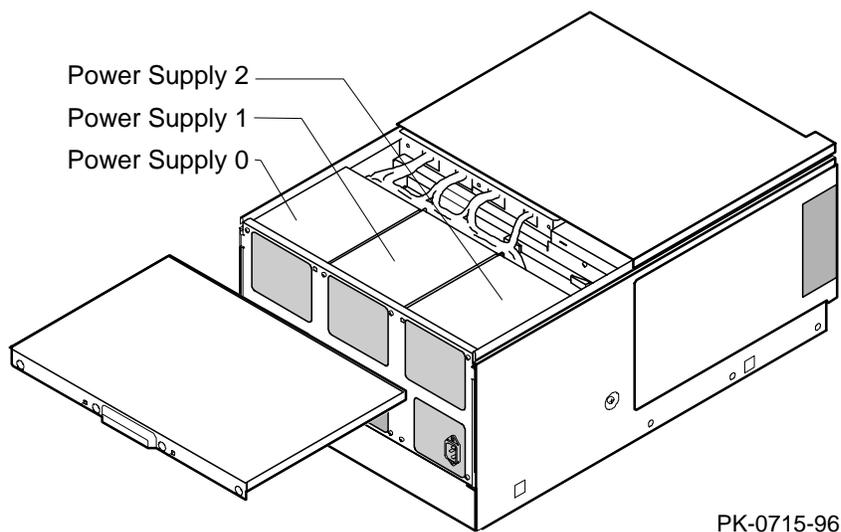
PK-0715-96

Description

One to three power supplies provide power to components in the system drawer. (They supply power only for the drawer in which they are located.) Three power supplies provide redundant power in fully loaded AlphaServer 4000/4100 systems.

These power supplies share the load, and redundant configurations are supported. They autoselect line voltage (120V to 240V). Each has 450 W output and supplies up to 75A of 3.43V, 50A of 5.0V, 11A of 12V, and small amounts of –5V, –12V, and auxiliary voltage (Vaux).

NOTE: The LEDs on some modules are on when the line cord is plugged in, regardless of the position of the On/Off button.

Configuration

- An AlphaServer 4100 system with one or two CPUs requires one power supply (two for redundancy).

- An AlphaServer 4100 system with three or four CPUs requires two power supplies (three for redundancy).

- An AlphaServer 4000 system with one or two CPUs and one PCI card cage requires one power supply (two for redundancy).

- An AlphaServer 4000 system with one or two CPUs and two PCI card cages requires two power supplies (three for redundancy).

- Power supply 0 is installed first, power supply 2 second, and power supply 1 third. See Figure 1-22. (The power supply numbering shown here corresponds to the numbering displayed by the SRM console's **show power** command.)

Chapter 2

Power-Up

This chapter describes system power-up testing and explains the power-up displays. The following topics are covered:

- Control Panel
- Power-Up Sequence
- SROM Power-Up Test Flow
- SROM Errors Reported
- XSROM Power-Up Test Flow
- XSROM Errors Reported
- Console Power-Up Tests
- Console Device Determination
- Console Power-Up Display
- Fail-Safe Loader

2.1 Control Panel

The control panel display indicates the likely device when testing fails.

Figure 2-1   Control Panel and LCD Display
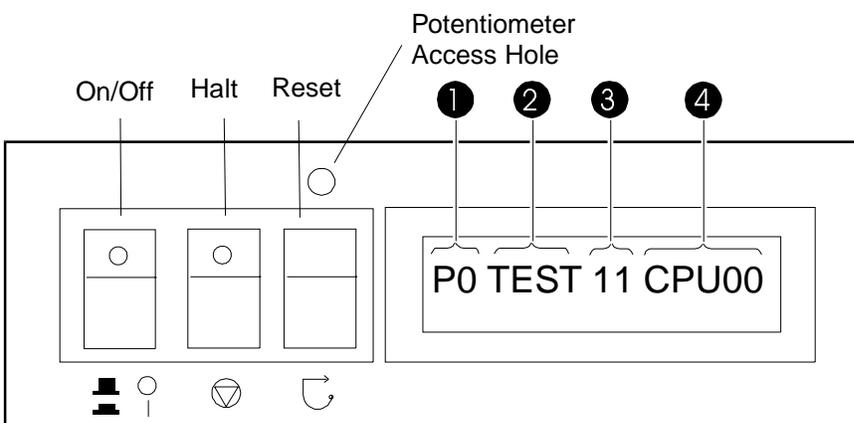
PK-0706G-96

- When the On/Off button LED is on, power is applied and the system is running. When it is off, the system is not running, but power may or may not be present. If power is present, the PCM or the power LED on the system bus to PCI bus bridge module should be flashing. Otherwise, there is a power problem.

- When the Halt button LED is lit and the On/Off button is on, the system should be running either the SRM console or Windows NT. If the Halt button is in, but the LED is off, the OCP, its cables, or the PCM is likely to be broken.

Table 2–1   Control Panel Display

Field	Content	Display	Meaning
❶	CPU number	P0–P3	CPU reporting status
❷	Status	TEST	Tests are executing
		FAIL	Failure has been detected
		MCHK	Machine check has occurred
		INTR	Error interrupt has occurred
❸	Test number		
❹	Suspected device	CPU0–3	CPU module number[1]
		MEM0–3 and L, H, or *	Memory pair number and low module, high module, or either[2]
		IOD0	Bridge to PCI bus 0[3]
		IOD1	Bridge to PCI bus 1[3]
		IOD2	Bridge to PCI bus 2[4]
		IOD3	Bridge to PCI bus 3[4]
		FROM0	Flash ROM[5]
		COMBO	COM controller[5]
		PCEB	PCI-to-EISA bridge[5]
		ESC	EISA system controller[5]
		NVRAM	Nonvolatile RAM[5]
		TOY	Real-time clock[5]
		I8242	Keyboard and mouse controller[5]

The potentiometer, accessible through the access hole just above the Reset button controls the intensity of the LCD. Use a small Phillips head screwdriver to adjust.

---

[1] CPU module

[2] Memory module

[3] Bridge module (B3040-AA)

[4] Bridge module (B3040-AB)

[5] EISA/PCI motherboard

2.2 Power-Up Sequence

Console and most power-up tests reside on the I/O subsystem, not on the CPU nor on any other module on the system bus.

Figure 2-2   Power-Up Flow
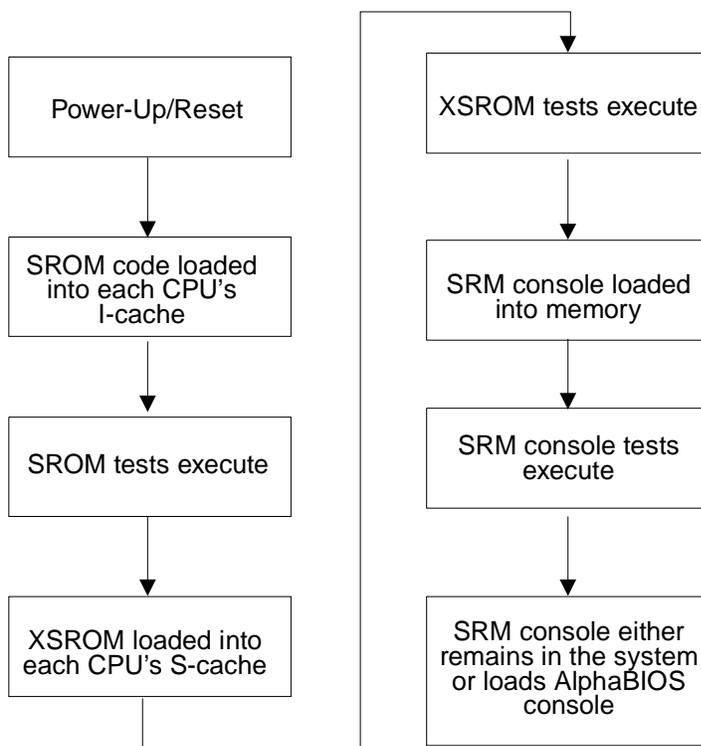
PKW0432B-96

Definitions

SROM. The SROM is a 128-Kbit ROM on each CPU module. The ROM contains minimal diagnostics that test the Alpha chip and the path to the XSROM. Once the path is verified, it loads XSROM code into the Alpha chip and jumps to it.

XSROM. The XSROM, or extended SROM, contains back-up cache and memory tests, and a fail-safe loader. The XSROM code resides in sector 0 of FEPROM 0 on

the XBUS. Sector 2 of FEPROM 0 contains a duplicate copy of the code and is used if sector 0 is bad.

FEPROM. Two 1-Mbyte programmable ROMs are on the XBUS on PCI0. FEPROM 0 contains two copies of the XSROM, the OpenVMS and DIGITAL UNIX PALcode, and the SRM console and decompression code. FEPROM 1 contains the AlphaBIOS and NT HALcode. See Figure 2-3. These two FEPROMs can be flash updated. Refer to Appendix A.

Figure 2-3   Contents of FEPROMs
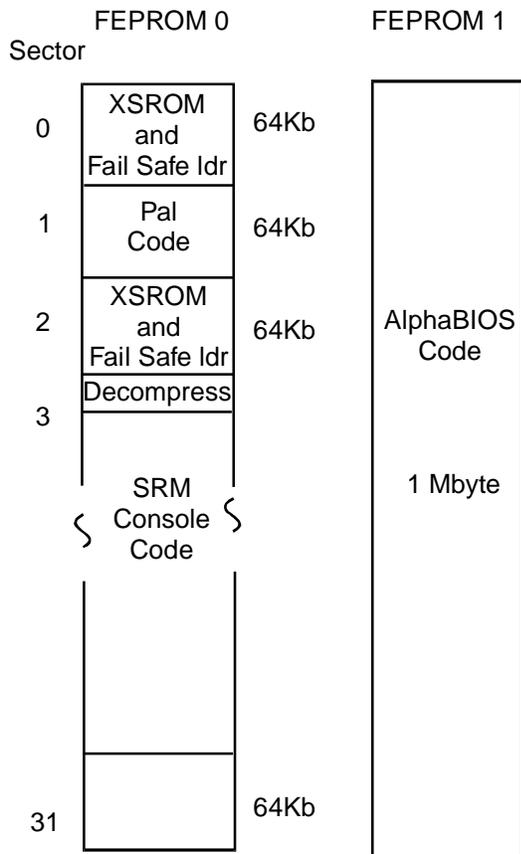
PKW0431D-96

For the console to run, the path from the CPU to the XSROM must be functional. The XSROM resides in FEPROM0 on the XBUS, off the EISA bus, off PCI 0, off IOD 0. See Figure 2-4. This path is minimally tested by SROM.

Figure 2-4   Console Code Critical Path (4100 Block Diagram)
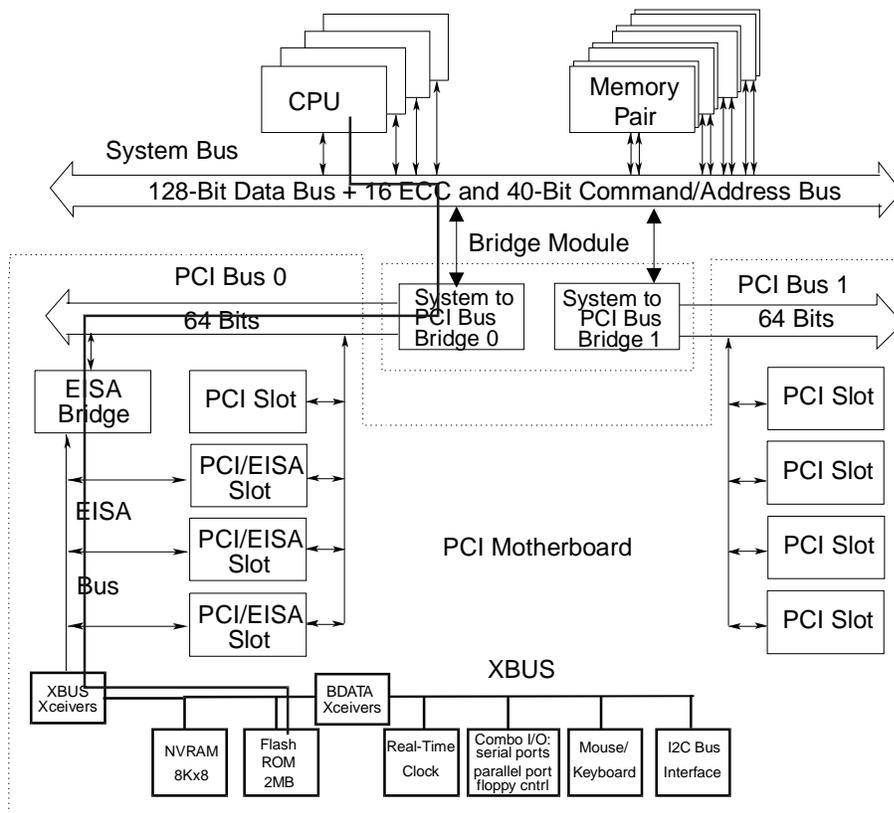
PKW0431E-96

The SROM contents are loaded into each CPU's I-cache and executed on power-up/reset. After testing the caches on each processor chip, it tests the path to the XSROM. Once this path is tested and deemed reliable, layers of the XSROM are loaded sequentially into the processor chip on each CPU. None of the SROM or XSROM power-up tests are run from memory—all run from the caches in the CPU chip, thus providing excellent diagnostic isolation. Later power-up tests, run under the console, are used to complete testing of the I/O subsystem.

There are two console programs: the SRM console and the AlphaBIOS console, as detailed in the *AlphaServer 4100 System Drawer User's Guide* (EK–4100A–UG) and the *AlphaServer 4000 System Drawer User's Guide* (EK–4000A–UG). By default, the SRM console is always loaded and I/O system tests are run under it before the system loads AlphaBIOS. To load AlphaBIOS, the **os_type** environment variable must be set to **NT** and the Halt button should be out (LED not lit). Otherwise, the SRM console continues to run.

2.3 SROM Power-Up Test Flow

The SROM tests the CPU chip and the path to the XSROM.

Figure 2-5   SROM Power-Up Test Flow

PKW0432-96
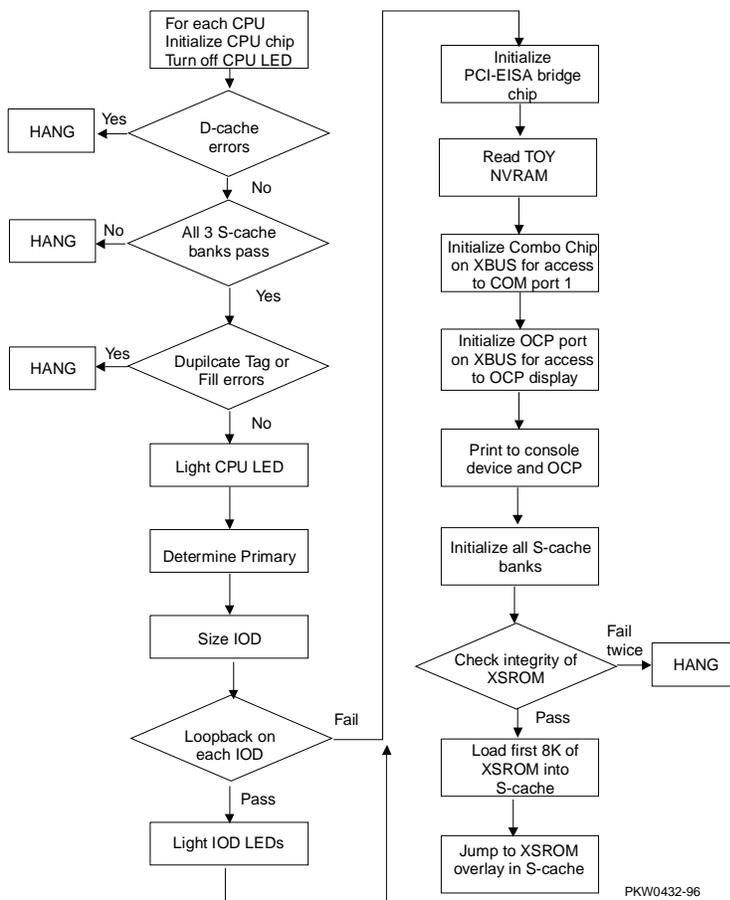
The Alpha chip built-in self-test tests the I-cache at power-up and upon reset.

Each CPU chip loads its SROM code into its I-cache and starts executing it. If the chip is partially functional, the SROM code continues to execute. However, if the chip cannot perform most of its functions, that CPU hangs and that CPU pass/fail LED remains off.

If the system has more than one CPU and at least one passes both the SROM and XSROM power-up tests, the system will bring up the console. The console checks the FW_SCRATCH register where evidence of the power-up failure is left. Upon finding the error, the console sends these messages to COM1 and the OCP:

- COM1 (or VGA): Power-up tests have detected a problem with your system
- OCP: Power-up failure

Table 2-2 lists the tests performed by the SROM.

Table 2-2   SROM Tests

Test Name	Logic Tested
D-cache RAM March test	D-cache access, D-cache data, D-cache address logic
D-cache Tag RAM March test	D-cache tag store RAM, D-cache bank address logic
S-cache Data March test	S-cache RAM cells, S-cache data path, S-cache address path
S-cache Tag RAM March test	S-cache tag store RAM, S-cache bank address logic
I-cache Parity Error test	I-cache parity error detection, ISCR register and error forcing logic, IC_PERR_STAT register and reporting logic
D-cache Parity Error test	D-cache parity error detection, DC_MODE register and parity error forcing logic, DC_PERR_STAT register and reporting logic
S-cache Parity Error test	S-cache parity error detection, AC_CTL register and parity error forcing logic, SC_STAT register and reporting logic
IOD Access test	Access to IOD CSRs, data path through CAP chip and MDP0 on each IOD, PCI0 A/D lines <31:0>

2.4 SROM Errors Reported

The SROM reports machine checks, pending interrupt/exception errors, and
errors related to corruption of FEPROM 0. If SROM errors are fatal, the
particular CPU will hang and only the CPU self-test pass LEDs and/or the LEDs
on the system bus to PCI bus bridge module will indicate the failure.

Example 2-1   SROM Errors Reported at Power-Up

Unexpected Machine Check (CPU Error)

```
UNEX MCHK on CPU 0
EXC_ADR 42a9
EI_STAT ffffff004ffffff
EI_ADDR fffff000000801f
SC_STAT 0
SC_ADDR FFFFFF0000005F2F
```

Pending Interrupt/Exception (CPU Error)

```
INT-EXC on CPU0
ISR         400000
EI_STAT ffffff007ffffff
EI_ADDR ffffff7ffffffdf
FIL_SYN 631B
BCTGADR fffffa7fffcafff
```

FEPROM Failures (PCI Motherboard Error)

```
Sctr 0 -XSROM headr PTTRN fail
Sctr 0 -XSROM headr CHKSM fail
Sctr 0 -XSROM code CHKSM fail
Sctr 2 -XSROM headr PTTRN fail
Sctr 2 -XSROM headr CHKSM fail
Sctr 2 -XSROM code CHKSM fail
```

2.5 XSROM Power-Up Test Flow

Once the SROM has completed its tests and verified the path to the FEPROM containing the XSROM code, it loads the first 8 Kbytes of XSROM into the primary CPU's S-cache and jumps to it.

Figure 2-6 XSROM Power-Up Flowchart
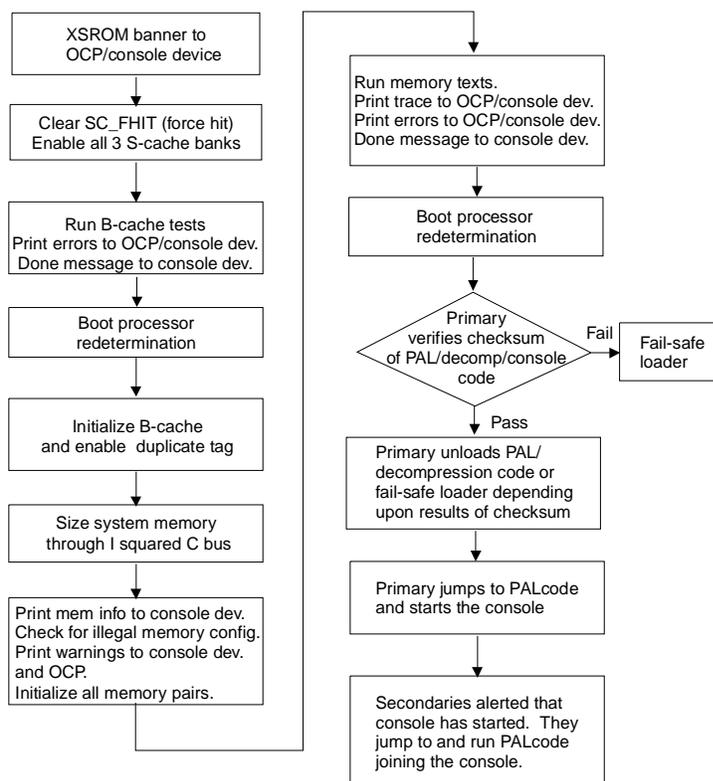
XSROM tests are described in Table 2-3. Failure indicates a CPU failure.

After jumping to the primary CPU's S-cache, the code then intentionally I-caches itself and is completely register based (no D-stream for stack or data storage is used). The only D-stream accesses are writes/reads during testing.

Each FEPROM has sixteen 64-Kbyte sectors. The first sector contains B-cache tests, memory tests, and a fail-safe loader. The second sector contains PALcode. The third sector contains a copy of the first sector. The remaining thirteen sectors contain the SRM console and decompression code.

NOTE: Memory tests are run during power-up and reset (see Table 2-4). They are also affected by the state of the memory_test environment variable, which can have the following values:

FULL	Test all memory
PARTIAL	Test up to the first 256 Mbytes
NONE	Test 32 Mbytes

Table 2-3   XSROM Tests

Test	Test Name	Logic Tested
11	B-cache Tag Data Line test	Access to B-cache tags, shorts between tag data and its status and parity bits
12	B-cache Tag March test	B-cache tag store RAMs, B-cache STAT store RAMs
13	B-cache Data Line test	B-cache data lines to B-cache data RAMs, B-cache read/write logic
14	B-cache Data March test	B-cache data RAMs, CPU chip B-cache control, CPU chip B-cache address decode, INDEX_H<2x:6> (address bus)
15	B-cache ECC Data Line test	CPU chip ECC generation and checking logic, ECC lines from CPU chip to B-cache, B-cache ECC RAMs
16	B-cache Data ECC March test	Portion of B-cache data RAMs used for ECC
17	CPU chip ECC Single/Double bit Error test	CPU chip ECC single-bit error detection and correction, ECC double-bit error detection, ECC error reporting
18	B-cache Tag Store Parity Error test	B-cache tag array, CPU parity detection, EI_ADDR and EI_STAT register operation
19	B-cache STAT Store Parity Error test	B-cache STAT array, CPU chip B-cache STAT parity generation/detection

Table 2-4   Memory Tests

Test	Test Name	Logic Tested	Description
20	Memory Data test	Data path to and from memory Data path on memory and RAMs	01 – FF Errors are reported as an 8-bit binary field. A set bit indicates a module failure. Bit <0> indicates pass/fail of MEM0_L; <1> indicates pass/fail of MEM0_H; <2> indicates pass/fail of MEM1_L; <7> indicates pass/fail of MEM3_H.
21	Memory Address test	Address path to and from memory Address path on memory and RAMs	Same as test 20.
23*	Memory Bitmap Building	No new logic	Maps out bad memory by way of the bitmap. It does not completely fail memory.
24	Memory March test	No new logic	Maps out bad memory.

* There is no test 22.

2.6 XSROM Errors Reported

The XSROM reports B-cache test errors and memory test errors. It also reports a warning if memory is illegally configured.

Example 2-2   XSROM Errors Reported at Power-Up

B-cache Error (CPU Error)

```
TEST ERR on cpu0                        #CPU running the test
FRU          cpu0
err#   2
tst#   11
exp:   5555555555555555                  #Expected data
rcv:   aaaaaaaaaaaaaaaa                  #Received data
adr:   ffff8                             #B-cache location error
                                         #occurred
```

Memory Error (Memory Module Indicated)

```
20..21..
TEST ERR on cpu0                 #CPU running test
FRU:  MEM1L                      #Low member of memory pair 1

err#  c
tst#  21
22..23..24..Memory testing complete on cpu0
```

Memory Configuration Error (Operator Error)

```
ERR!   mem_pair0 misconfigured

ERR!   mem_pair1 card size mismatch

ERR!   mem_pair1 card type mismatch

ERR!   mem_pair1 EMPTY
```

FEPROM Failures (PCI Motherboard Error)

```
Sctr 1 -PAL headr PTTRN fail
Sctr 1 -PAL headr CHKSM fail
Sctr 1 -PAL code CHKSM fail
Sctr 3 -CONSLE headr PTTRN fail
Sctr 3 -CONSLE headr CHKSM fail
Sctr 3 -CONSLE code CHKSM fail
```

2.7 Console Power-Up Tests

Once the SRM console is loaded, it does further testing of each IOD. Table 2-5 describes the IOD power-up tests, and Table 2-6 describes the PCI motherboard power-up tests.

Table 2-5 IOD Tests

Test Number	Test Name	Description
1	IOD CSR Access test	Read and write all CSRs in each IOD.
2	Loopback test	Dense space writes to the IOD's PCI dense space to check the integrity of ECC lines on the IODs.
3	ECC test	Loopback tests similar to test 2 but with a varying pattern to create an ECC of 0s. Single- and double-bit errors are checked.
4	Parity Error and Fill Error tests	Parity errors are forced on the address and data lines on system bus and PCI buses. A fill error transaction is forced on the system bus.
5	Translation Error test	A loopback test using scatter/gather address translation logic on each IOD.
6	Write Pending test	Runs test 2 with the write-pending bit set and clear in the CAP chip control register.
7	PCI Loopback test	Loops data through each PCI on each IOD, testing the mask field of the system bus.
8	PCI Peer-to-Peer Byte Mask test	Tests that devices on the same PCI and on different PCIs can communicate.

Table 2-6 PCI Motherboard Tests (B3050 only)

Test Number	Test Name	Diagnostic Name	Description
1	PCEB	pceb_diag	Tests the PCI to EISA bridge chip
2	ESC	esc_diag	Tests the EISA system controller
3	8K NVRAM	nvram_diag	Tests the NVRAM
4	Real-Time Clock	ds1287_diag	Tests the real-time clock chip
5	Keyboard and Mouse	i8242_diag	Tests the keyboard/mouse chip
6	Flash ROM	flash_diag	Dumps contents of flash ROM
7	Serial and Parallel Ports and Floppy	combo_diag	Tests COM ports 1 and 2, the parallel port, and the floppy
8	CD-ROM	ncr810_diag	Tests the CD-ROM controller

For both IOD tests and PCI 0 and PCI 1 tests, trace and failure status is sent to the OCP. If any of these tests fail, a warning is sent to the SRM console device after the console prompt (or AlphaBIOS pop-up box). The LEDs on the system bus to PCI bus bridge module are controlled by the diagnostics. If a LED is off, a failure occurred.

2.8 Console Device Determination

After the SROM and XSROM have completed their tasks, the SRM console
program, as it starts, determines where to send its power-up messages.

Figure 2-7   Console Device Determination Flowchart

```
          ┌─────────────────────┐
          │   Power-Up/Reset     │
          │         or           │
          │   P00>>>   Init      │
          └─────────────────────┘
                     │
                     ▼
              ╱──────────────╲        No      ╱──────────────╲
             ╱  Console Envar  ╲───────────▶ ╱  Console Envar  ╲
             ╲   = serial      ╱             ╲   = graphics     ╱
              ╲──────────────╱                ╲──────────────╱
                     │ Yes                          │ Yes
                     ▼                              ▼
          ┌─────────────────────┐          ╱──────────────╲     Yes    ┌─────────────────────┐
          │  Enable COM port 1   │         ╱  VGA adapter   ╲──────────▶│ VGA becomes the      │
          │  and send messages   │         ╲     on          ╱         │ console device.      │
          │ as system is powering up│       ╲    PCI0       ╱           └─────────────────────┘
          └─────────────────────┘           ╲──────────────╱
                                                   │ No
                                                   ▼
                                    ┌─────────────────────────────┐
                                    │     Enable COM port 1        │
                                    │     and send messages        │
                                    │  as system is powering up.   │
                                    │  Warning message sent if a   │
                                    │ VGA adapter is seen on PCI 1 │
                                    └─────────────────────────────┘
```
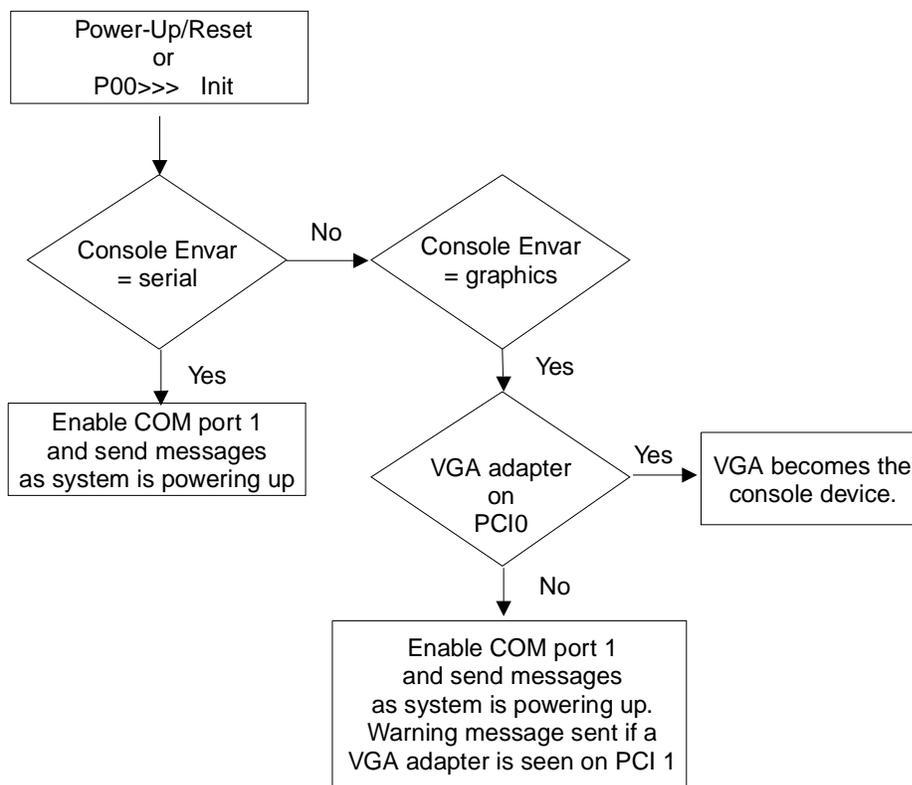
PKW0434-96

Console Device Options

The console device can be either a serial terminal or a graphics monitor. Specifically:

- A serial terminal connected to COM1 off the server control module. The terminal connected to COM1 must be set to 9600 baud. This baud rate cannot be changed.

- A graphics monitor off an adapter on PCI0.

Systems running Windows NT must have a graphics monitor as the console device and run AlphaBIOS as the console program.

During power-up, the SROM and the XSROM always send progress and error messages to the OCP and to the COM1 serial port if the SRM **console** environment variable (set with the **set console** command) is set to **serial**. If the **console** environment variable is set to **graphics**, no messages are sent to COM1.

If the console device is connected to COM1, the SROM, XSROM, and console power-up messages are sent to it once it has been initialized. If the console device is a graphics device, console power-up messages are sent to it, but SROM and XSROM power-up messages are lost. No matter what the **console** environment variable setting, each of the three programs sends messages to the control panel display.

Messages Sent By	Console Set to	
	Serial	Graphics
SROM	COM1	Lost
XSROM	COM1	Lost
SRM console	COM1	VGA

Changing Where Console Output Is Displayed

You can change where console output is displayed, assuming the SRM console has fully powered up and the **os_type** environment variable is set to **openvms** or **unix**. (The following does not work if **os_type** is set to **nt**.)

If the **console** environment variable is set to **serial** and no serial terminal is attached to COM1, pressing a carriage return on a graphics monitor attached to the system makes it the console device and the console prompt is sent to it. If the console environment variable is set to **graphics** and no graphics monitor is attached to the adapter, pressing a carriage return on a serial terminal attached to COM1 makes it the console device and the console prompt is sent to it.

2.9 Console Power-Up Display

The entire power-up display prints to a serial terminal (if the *console* environment variable is set to *serial*), and parts of it print to the control panel display. The last several lines print to either a serial terminal or a graphics monitor.

Example 2-3   Power-Up Display

```
 SROM V1.0 on cpu0                                          ❶
 SROM V1.0 on cpu1
 SROM V1.0 on cpu2
 SROM V1.0 on cpu3
XSROM V1.0 on cpu2                                          ❷
XSROM V1.0 on cpu1
XSROM V1.0 on cpu3
XSROM V1.0 on cpu0
BCache testing complete on cpu2                            ❸
BCache testing complete on cpu0
BCache testing complete on cpu3
BCache testing complete on cpu1
mem_pair0 - 128 MB                                         ❹
mem_pair1 - 128 MB
20..20..21..20..21..20..21..21..23..24..24..24..24..❺
Memory testing complete on cpu0
Memory testing complete on cpu1
Memory testing complete on cpu3
Memory testing complete on cpu2
```

❶ At power-up or reset, the SROM code on each CPU module is loaded into that module's I-cache and tests the module. If all tests pass, the processor's LED lights. If any test fails, the LED remains off and power-up testing terminates on that CPU.

The first determination of the primary processor is made, and the primary processor executes a loopback test to each PCI bridge. If this test passes, the bridge LED lights. If it fails, the LED remains off and power-up continues. The EISA system controller, PCI-to-EISA bridge, COM1 port, and control panel port are all initialized thereafter.

Each CPU prints an SROM banner to the device attached to the COM1 port and to the control panel display. (The banner prints to the COM1 port if the **console** environment variable is set to **serial**. If it is set to **graphics**, nothing prints to the console terminal, only to the control panel display, until ❻).

❷ Each processor's S-cache is initialized, and the XSROM code in the FEPROM on the PCI 0 is unloaded into them. (If the unload is not successful, a copy is unloaded from a different FEPROM sector. If the second try fails, the CPU hangs.)

Each processor jumps to the XSROM code and sends an XSROM banner to the COM1 port and to the control panel display.

❸ The three S-cache banks on each processor are enabled, and then the B-cache is tested. If a failure occurs, a message is sent to the COM1 port and to the control panel display.

Each CPU sends a B-cache completion message to COM1.

❹ The primary CPU is again determined, and it sizes memory by reading memory registers on the I^2C bus.

The information on memory pairs is sent to COM1. If an illegal memory configuration is detected, a warning message is sent to COM1 and the control panel display.

❺ Memory is initialized and tested, and the test trace is sent to COM1 and the control panel display. Each CPU participates in the memory testing. The numbers for tests 20 and 21 might appear interspersed, as in Example 2-3. This is normal behavior. Test 24 can take several minutes if the memory is very large. The message "P0 TEST 24 MEM**" is displayed on the control panel display; the second asterisk rotates to indicate that testing is continuing. If a failure occurs, a message is sent to the COM1 port and to the control panel display.

Each CPU sends a test completion message to COM1.

Example 2-3   Power-Up Display (Continued)

```
starting console on CPU 0                                    ❻
sizing memory                                                ❼
   0    128 MB SYNC
   1    128 MB SYNC
starting console on CPU 1
starting console on CPU 2
starting console on CPU 3
probing IOD1 hose 1                                          ❽
   bus 0 slot 1 - NCR 53C810
   bus 0 slot 2 - DECchip 21041-AA
   bus 0 slot 3 - NCR 53C810
   bus 0 slot 4 - DECchip 21040-AA
probing IOD0 hose 0
   bus 0 slot 1 - PCEB
Configuring I/O adapters...
AlphaServer 4100 Console V1.0, 13-MAR-1996 18:18:26 ❾
P00>>>
```

❻ The final primary CPU determination is made. The primary CPU unloads PALcode and decompression code from the FEPROM on the PCI 0 to its B-cache. The primary CPU then jumps to the PALcode to start the SRM console.

The primary CPU prints a message indicating that it is running the console. Starting with this message, the power-up display is printed to the default console terminal, regardless of the state of the **console** environment variable. (If **console** is set to **graphics**, the display from here to the end is saved in a memory buffer and printed to the graphics monitor after the PCI buses are sized and the graphics device is initialized.)

❼ The size and type of each memory pair is determined.

The console is started on each of the secondary CPUs. A status message prints for each CPU.

❽ The PCI bridges (indicated as IOD*n*) are probed and the devices are reported. I/O adapters are configured.

❾ The SRM console banner and prompt are printed. (The SRM prompt is shown in this manual as P00>>>. It can, however, be P01>>>, P02>>>, or P03>>>. The number indicates the primary processor.) If the **auto_action** environment variable is set to **boot** or **restart** and the **os_type** environment variable is set to **unix** or **openvms**, the DIGITAL UNIX or OpenVMS operating system boots.

If the system is running the Windows NT operating system (the **os_type** environment variable is set to **nt**), the SRM console loads and starts the AlphaBIOS console and does not print the SRM banner or prompt.

2.10 Fail-Safe Loader

The fail-safe loader is a software routine that loads the SRM console image from floppy. Once the console is running you will want to run LFU to update FEPROM 0 with a new image.

NOTE: FEPROM 0 contains images of the SROM, XSROM, PAL, decompression, and SRM console code.

If the fail-safe loader loads, the following conditions exist on the machine:

- The SROM has passed its tests and successfully unloaded the XSROM. If the SROM fails to unload both copies of XSROM, it reports the failure to the control panel display and COM1 if possible, and the system hangs.

- The XSROM has completed its B-cache and memory tests but has failed to unload the PALcode in FEPROM 0 sector 1 or the SRM console code.

- The XSROM reports the errors encountered and loads the fail-safe loader.

Chapter 3

Troubleshooting

This chapter describes troubleshooting during power-up and booting, as well as diagnostics for AlphaServer 4000/4100 systems. The following topics are covered:

- Troubleshooting with LEDs
- Troubleshooting Power Problems
- Running Diagnostics—Test Command

3.1 Troubleshooting with LEDs

During power-up, reset, initialization, or testing, diagnostics are run on CPUs, memories, bridge modules, PCI motherboards, and sometimes options. The following sections describe possible problems that can be identified by checking LEDs.

Figure 3-1   CPU and Bridge Module LEDs

Bridge Module LEDs (IOD 0 & 1)   CPU LEDs

IOD0 Self-Test Pass
IOD1 Self-Test Pass

DC_OK
SROM Oscillator
CPU Self-Test Pass
Regulator OK (EV56)

POWER_FAN_OK
TEMP_OK

Bridge Module LEDs (IOD 2 & 3)

Normally On
Normally Off

IOD2 Self-Test Pass
IOD3 Self-Test Pass

PKW0400C-97
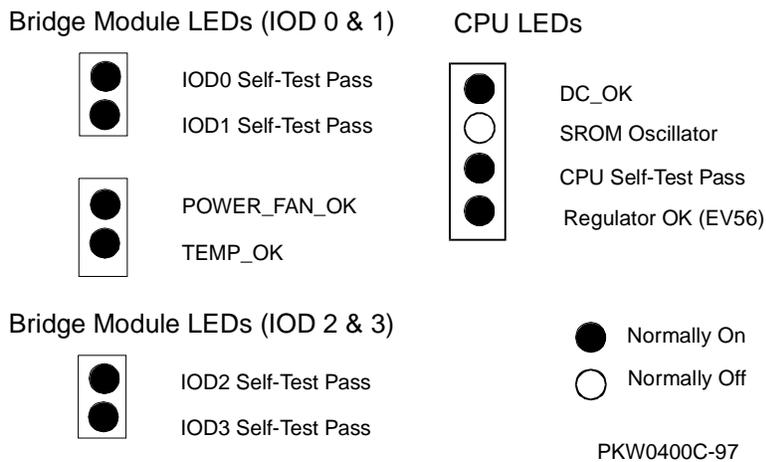
CPU LEDs

- If the CPU STP LED on any CPU module is lit, that CPU chip is functioning properly. If the operating system is NT and the CPU STP LED is off, that CPU may or may not be functioning.

  You can use the Halt button on the OCP to prevent the AlphaBIOS console (which turns off the CPU STP LED) from booting, thus assuring the validity of the CPU STP LED. If the LED is off, replace the CPU. If the LED is lit, you can use the SRM console command **alphabios** to load and run the AlphaBIOS console.

- The top LED on a CPU module is a DC OK LED. It is driven by the PCM module. If it is not lit, there are probably power problems.

- The second from the top LED on a CPU lights only when the SROM on the CPU is loaded.

- On modules with EV56 CPU processors a fourth LED is present at the bottom of the column. The LED is normally on indicating that the power regulator on the module is working properly. If the LED is off, replace the module.

System Bus to PCI Bus Bridge Module LEDs (B3040-AA)

There are four LEDs on the B3040-AA system bus to PCI bus bridge module:

- The top two LEDs indicate the condition of the bridge module. If either is off, the module should be replaced.

- The bottom two LEDs are passed from the PCM. Both should be on during normal operation. If either is off while the system is on, the LEDs on the PCM module should indicate what failed. If they do not, the PCM could be broken or the bridge module is not passing the signals to the LEDs.

  NOTE: If AC power is applied and the system is off and a power supply is in operation, the power LED, the top one of the bottom two, flashes, indicating the presence of Vaux (auxiliary voltage).

System Bus to PCI Bus Bridge Module LEDs (B3040-AB)

There are two LEDs on the B3040-AB system bus to PCI bus bridge module:

- The two LEDs indicate the condition of the bridge module. If either is off, the module should be replaced.

3.1.1 Cabinet Power and Fan LEDs

Figure 3-2 Cabinet Power and Fan LEDs

Fan LED

Power LED

PK-0664-96
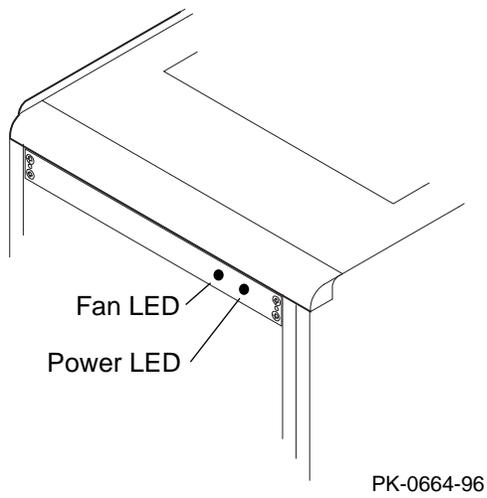
A cabinet system has three exhaust fans at the top of the cabinet. They are powered from a small power supply in the fan tray. This power supply also powers the server control module at the bottom of the PCI card cage to allow remote access to the system. A failure of the power supply is indicated only by the LEDs. No messages are displayed.

There are two LEDs on the top panel: a fan LED and a power LED.

- When the fan LED (amber) is flashing, a cabinet fan needs replacing. Look to see which fan appears broken (either not functioning at all; or it appears to be slower than the others).

- When the power LED (green) is off, either the power supply in the fan tray is broken or there is a power problem.

3.2 Troubleshooting Power Problems

Power problems can occur before the system is up or while the system is running. If a system stops running, make a habit of checking the PCM.

Power Problem List

The system will halt for the following:

1. A CPU fan failure
2. A system fan failure
3. An overtemperature condition
4. Power supplied out of tolerance
5. Circuit breaker(s) tripped
6. AC problem
7. Interlock switch activation or failure
8. PCM failure
9. Environmental electrical failure or
   unrecoverable system fault
   with auto_action ev = halt or boot
10. Operator error - failure to unplug all power
    supplies and letting Vaux drain (10 sec delay)
    before restarting
11. Cable failure
12. Module failure - System motherboard, PCI
    motherboard, or system bus to PCI bus bridge
13. SCM breaking the interlock circuit

Indications of failure:

1. Power control module LEDs indicate
   CPU fan, system fan, overtemperature,
   and power supply failures
2. Circuit breaker(s) tripped

No obvious indications for failures 7 - 13
from the power system.

PKW0436A-96

If Halt Is Caused by Power, Fan, or Overtemperature

If a system is stopped because of a power, fan, or overtemperature problem, use the
PCM LEDs to diagnose the problem. See Section 3.2.1.

If Power Problem Occurs at Power-Up

If the system has a power problem on a cold start, the PCM LEDs are not valid until after DCOK_SENSE has been asserted. The cause is one of the following:

- Broken system fan

- Broken CPU fan

- Power supplied to the system is out of tolerance (a power supply could be broken and the system could still power up)

- PCM failure

- Interlock failure

- Wire problems

- Temperature problem (unlikely)

Recommended Order for Troubleshooting Failure at Power-Up

1. Check to see if any CPU fan or system fan is not spinning. Fans can fail by not spinning and/or not putting out the tachometer output necessary as input to the PCM comparator that checks the fans. (See steps 4 and 5.) Replace broken fan.

2. Replace the PCM.

3. Sequentially remove CPUs and try to power up after you remove a CPU. If the system powers up, the last CPU you removed had a fan failure.

4. Check the output of the power supplies. See Section 4.1 for locations of +5 and +3.43 volt output pins. If the output is above or below the threshold, replace the faulty power supply.

5. Check the output of each system fan with a voltmeter. Probe the middle of three outputs of the fans with the positive lead of the meter and ground the other probe. The meter should read 2.5 volts to 3 volts. If a fan's output is out of this range, replace the fan.

   NOTE: You will have to disable the interlocks to check the voltages in step 5. You will have only 10 seconds to measure them. There is a 10-second delay before the PCM turns off the power.

   The PCM must sense a change in Vaux (auxiliary voltage) to start the power supplies. Pressing the On button has no effect if the machine halted because of a failure in the power system. The power supplies must be unplugged and plugged back in for the On button to work.

3.2.1 Power Control Module LEDs

The PCM has 11 LEDs visible through the system card cage. The LED display shows the relative placement of the LEDs.

Figure 3-3   PCM LEDs

DCOK_SENSE
PS0_OK
PS1_OK
PS2_OK
TEMP_OK
CPUFAN_OK
SYSFAN_OK
CS_FAN0
CS_FAN1
CS_FAN2
C_FAN3

Normally On

Tested at one-second intervals

Off if power supply not present or broken
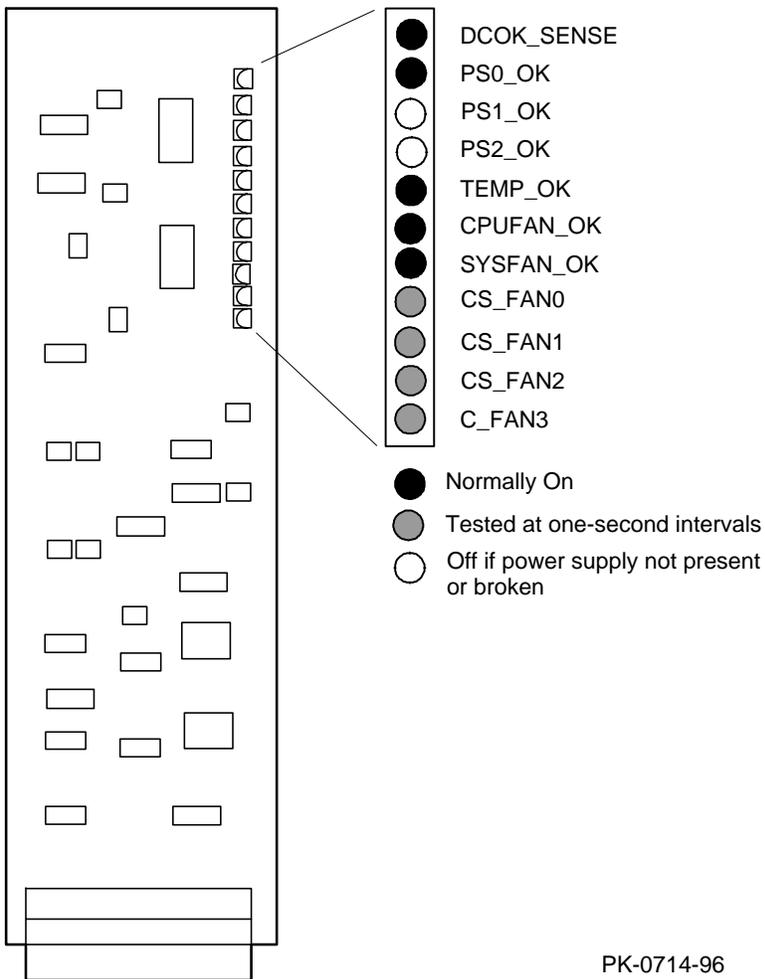
PK-0714-96

Table 3-1   Power Control Module LED States

LED	State	Description
DCOK_SENSE	On	Both +5.0V and +3.43V are present and within limits.
PS0_OK	On	Power supply 0 is present and has asserted POK_H.
PS1_0K	On Off	Power supply 1 is present and has asserted POK_H. Power supply 1 not present.
PS2_OK	On Off	Power supply 2 is present and has asserted POK_H. Power supply 2 not present.
TEMP_OK	On	The system temperature is below 55° C.
CPUFAN_OK	On Off	All CPU fans are OK. A CPU fan has failed. The specific fan is identified by the CS_FANx or C_FAN3 LED that remains lit.
SYSFAN_OK	On Off	All system fans are OK. A system fan has failed. The specific fan is identified by the CS_FANx that remains lit.
CS_FAN0	On Off	CPU fan 0 and system fan 0 are being sampled or one of them has failed as indicated by CPUFAN_OK and SYSFAN_OK. CPU fan 0 and system fan 0 are not being sampled and are functioning properly.
CS_FAN1	On Off	CPU fan 1 and system fan 1 are being sampled or one of them has failed as indicated by CPUFAN_OK and SYSFAN_OK. CPU fan 1 and system fan 1 are not being sampled and are functioning properly.
CS_FAN2	On Off	CPU fan 2 and system fan 2 are being sampled or one of them has failed as indicated by CPUFAN_OK and SYSFAN_OK. CPU fan 2 and system fan 2 are not being sampled and are functioning properly.
C_FAN3	On Off	CPU fan 3 is being sampled or has failed as indicated by CPUFAN_OK and SYSFAN_OK. Off CPU fan 3 and system fan 3 are not being sampled and are functioning properly.

3.3 Maintenance Bus (I^{2}C Bus)

The I^{2}C bus (referred to as the "I squared C bus") is a small internal maintenance bus used to monitor system conditions scanned by the power control module, write the fault display, store error state, and track configuration information in the system. Although all system modules (not I/O modules) sit on the maintenance bus, only the I^{2}C controller accesses it. Everything written or read on the I^{2}C bus is done by the controller. The block diagram below notes differences between the AlphaServer 4000 and 4100 with respect to the I^{2}C bus.

Figure 3-4   I^{2}C Bus Block Diagram
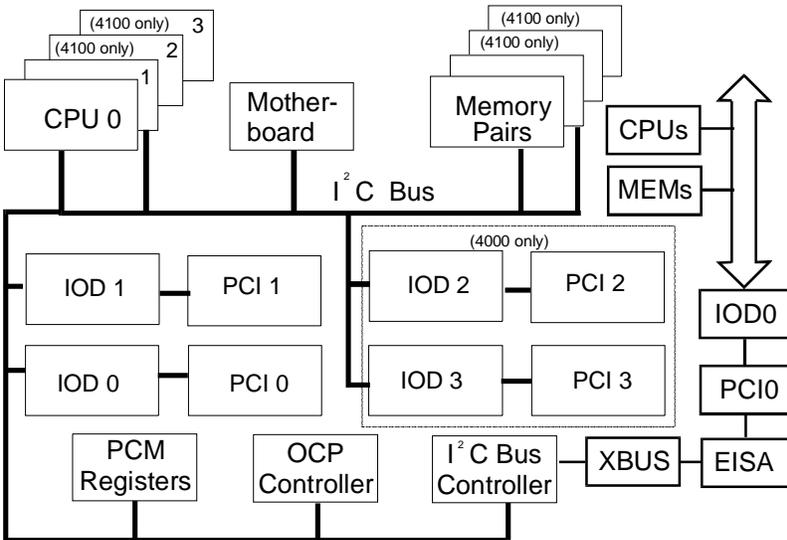
PKW0421F-97

Monitor

The I^{2}C bus monitors the state of system conditions scanned by the PCM. There are two registers on the PCM:

- One records the state of the fans and power supplies and is latched when there is a fault.

- The other causes an interrupt on the I^{2}C bus when a CPU or system fan fails, an overtemperature condition exists, or power supplied to the system is out of tolerance.

The interrupt received by the I^{2}C bus controller on PCI 0 alerts the system of imminent power shutdown. The controller has 30 seconds to read the two registers and store the information in the EEPROM on the PCM. The SRM console command **show power** reads these registers.

Fault Display

The OCP display is written through the I^{2}C bus.

Error State

Error state is written and read for power conditions. The state of the Halt button (in/out) is read on the I^{2}C bus.

Configuration Tracking

Each CPU, PCI bridge, PCI motherboard, and system motherboard has an EEPROM that contains information about the module that can be written and read over the I^{2}C bus. All modules contain the following information:

- Module type

- Module serial number

- Hardware revision

- Firmware revision

- Memory size (only required for memory modules)

3.4 Running Diagnostics — Test Command

The test command runs diagnostics on the entire system, CPU devices, memory devices, and the PCI I/O subsystem. The test command runs only from the SRM console. Ctrl/C stops the test.

Example 3-1   Test Command Syntax

```
P00>>> help test
FUNCTION


SYNOPSIS
  test ([-q] [-t <time>] [option]
  where option is:
  cpun
  memn
  pcin


  and n can be one of 0, 1, 2, 3, or *.


  The entire system is tested by default if no option specified.
```

NOTE: If you are running the Microsoft Windows NT operating system, switch from AlphaBIOS to the SRM console in order to enter the **test** *command. From the AlphaBIOS console, press in the Halt button (the LED will light) and reset the system, or select* **DIGITAL UNIX (SRM)** *or* **OpenVMS (SRM)** *from the* **Advanced CMOS Setup** *screen and reset the system.*

test [**-t** *time*] [**-q**] [*option*]

-t *time*   Specifies the run time in seconds. The default for system test is 600 seconds (10 minutes).

-q   Disables the display of status messages as exerciser processes are started and stopped during testing.

option   Either **cpu***n*, **mem***n*, or **pci***n*, where *n* is 0, 1, 2, 3, or *. If nothing is specified, the entire system is tested.

3.5 Testing an Entire System

A test command with no modifiers runs all exercisers for subsystems and devices on the system. I/O devices tested are supported boot devices. The test runs for 10 minutes.

Example 3-2   Sample Test Command

```
P00>>> test
Console is in diagnostic mode
System test, runtime 600 seconds

Type ^C to stop testing

Configuring system..
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1     DKa500                    RRD45  1645
polling ncr1 (NCR 53C810) slot 3, bus 0 PCI, hose 1   SCSI Bus ID 7
dkb200.2.0.3.1     DKb200                    RZ29B  0007
dkb400.4.0.3.1     DKb400                    RZ29B  0007
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0    DVA0                      RX23
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1: 08-00-2B-E5-B4-1A

Testing EWA0 network device

Testing VGA (alphanumeric mode only)

Starting background memory test, affinity to all CPUs..
Starting processor/cache thrasher on each CPU..
Starting processor/cache thrasher on each CPU..
Starting processor/cache thrasher on each CPU..
Starting processor/cache thrasher on each CPU..

Testing SCSI disks (read-only)
No CD/ROM present, skipping embedded SCSI test
Testing other SCSI devices (read-only)..

Testing floppy drive (dva0, read-only)
```

```
    ID       Program      Device      Pass  Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
00003047     memtest memory           1   0    0     134217728    134217728
00003050     memtest memory         205   0    0     213883392    213883392
00003059     memtest memory         192   0    0     200253568    200253568
00003062     memtest memory         192   0    0     200253568    200253568
00003084     memtest memory          80   0    0      82827392     82827392
000030d8    exer_kid dkb200.2.0.3    26   0    0             0     13690880
000030d9    exer_kid dkb400.4.0.3    26   0    0             0     13674496
0000310d    exer_kid dva0.0.0.100     0   0    0             0            0
    ID       Program      Device      Pass  Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
00003047     memtest memory           1   0    0     432013312    432013312
00003050     memtest memory         635   0    0     664716032    664716032
00003059     memtest memory         619   0    0     647940864    647940864
00003062     memtest memory         620   0    0     648989312    648989312
00003084     memtest memory         263   0    0     274693376    274693376
000030d8    exer_kid dkb200.2.0.3    90   0    0             0     47572992
000030d9    exer_kid dkb400.4.0.3    90   0    0             0     47523840
0000310d    exer_kid dva0.0.0.100     0   0    0             0       327680
    ID       Program      Device      Pass  Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
00003047     memtest memory           1   0    0     727711744    727711744
00003050     memtest memory        1054   0    0    1104015744   1104015744
00003059     memtest memory        1039   0    0    1088289024   1088289024
00003062     memtest memory        1041   0    0    1090385920   1090385920
00003084     memtest memory         447   0    0     467607808    467607808
000030d8    exer_kid dkb200.2.0.3   155   0    0             0     81488896
000030d9    exer_kid dkb400.4.0.3   155   0    0             0     81472512
0000310d    exer_kid dva0.0.0.100     1   0    0             0       607232
Testing aborted. Shutting down tests.
Please wait..

System test complete

^C
P00>>>
```

3.5.1 Testing Memory

**The test mem command tests individual memory devices or all memory. The
test shown in Example 3-3 runs for 2 minutes.**

Example 3-3   Sample Test Memory Command

```
P00>>> test memory
Console is in diagnostic mode
System test, runtime 120 seconds

Type ^C to stop testing

Starting background memory test, affinity to all CPUs..
Starting memory thrasher on each CPU..
Starting memory thrasher on each CPU..
Starting memory thrasher on each CPU..
Starting memory thrasher on each CPU..
```

ID	Program	Device	Pass	Hard/Soft		Bytes Written	Bytes Read
000046d7	memtest	memory	1	0	0	48234496	48234496
000046e0	memtest	memory	122	0	0	126862208	126862208
000046e9	memtest	memory	111	0	0	115329280	115329280
000046f2	memtest	memory	109	0	0	113232384	113232384
000046fb	memtest	memory	41	0	0	41937920	41937920
ID	Program	Device	Pass	Hard/Soft		Bytes Written	Bytes Read
000046d7	memtest	memory	1	0	0	226492416	226492416
000046e0	memtest	memory	566	0	0	592373120	592373120
000046e9	memtest	memory	555	0	0	580840192	580840192
000046f2	memtest	memory	554	0	0	579791744	579791744
000046fb	memtest	memory	211	0	0	220174080	220174080
ID	Program	Device	Pass	Hard/Soft		Bytes Written	Bytes Read
000046d7	memtest	memory	1	0	0	404750336	404750336
000046e0	memtest	memory	1011	0	0	1058932480	1058932480
000046e9	memtest	memory	1000	0	0	1047399552	1047399552
000046f2	memtest	memory	999	0	0	1046351104	1046351104
000046fb	memtest	memory	381	0	0	398410240	398410240

```
ID        Program      Device       Pass   Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
000046d7      memtest memory          1    0    0     583008256     583008256
000046e0      memtest memory       1456    0    0    1525491840    1525491840
000046e9      memtest memory       1446    0    0    1515007360    1515007360
000046f2      memtest memory       1444    0    0    1512910464    1512910464
000046fb      memtest memory        550    0    0     575597952     575597952
 ID        Program      Device       Pass   Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
000046d7      memtest memory          1    0    0     761266176     761266176
000046e0      memtest memory       1901    0    0    1992051200    1992051200
000046e9      memtest memory       1892    0    0    1982615168    1982615168
000046f2      memtest memory       1889    0    0    1979469824    1979469824
000046fb      memtest memory        720    0    0     753834112     753834112
 ID        Program      Device       Pass   Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
000046d7      memtest memory          1    0    0     937426944     937426944
000046e0      memtest memory       2346    0    0    2458610560    2458610560
000046e9      memtest memory       2337    0    0    2449174528    2449174528
000046f2      memtest memory       2333    0    0    2444980736    2444980736
000046fb      memtest memory        890    0    0     932070272     932070272

Memory test complete

Test time has expired...
P00>>>
```

3.5.2 Testing PCI

The test pci command tests PCI buses and devices. The test runs for 2 minutes.

Example 3-4 Sample Test Command for PCI

```
P00>>> test pci*
Console is in diagnostic mode
System test, runtime 120 seconds

Type ^C to stop testing

Configuring all PCI buses..
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1     DKa500                      RRD45  1645
polling ncr1 (NCR 53C810) slot 3, bus 0 PCI, hose 1   SCSI Bus ID 7
dkb200.2.0.3.1     DKb200                      RZ29B  0007
dkb400.4.0.3.1     DKb400                      RZ29B  0007
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1: 08-00-2B-E5-B4-1A
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0    DVA0                        RX23

Testing all PCI buses..

Testing EWA0 network device

Testing VGA (alphanumeric mode only)

Testing SCSI disks (read-only)

Testing floppy (dva0, read-only)
```

ID	Program	Device	Pass	Hard/Soft		Bytes Written	Bytes Read
00002c29	exer_kid	dkb200.2.0.3	27	0	0	0	14642176
00002c2a	exer_kid	dkb400.4.0.3	27	0	0	0	14642176
00002c5e	exer_kid	dva0.0.0.100	0	0	0	0	0

```
 ID        Program      Device        Pass  Hard/Soft Bytes Written  Bytes Read
-------- ------------ ------------ ------ --------- ------------- ------------
00002c29     exer_kid dkb200.2.0.3    92   0    0                0    48689152
00002c2a     exer_kid dkb400.4.0.3    92   0    0                0    48689152
00002c5e     exer_kid dva0.0.0.100     0   0    0                0      286720
Testing aborted. Shutting down tests.
Please wait..

Testing complete

^C
P00>>>
```

Chapter 4

Power System

This chapter describes the AlphaServer 4000/4100 power system:

- Power Supply
- Power Control Module Features
- Power Circuit and Cover Interlocks
- Power-Up/Down Sequencing
- Cabinet Power Configuration Rules
- Pedestal Power Configuration Rules (North America and Japan)
- Pedestal Power Configuration Rules (Europe and Asia Pacific)

4.1 Power Supply

Power supply ouputs are shown in Figure 4-1.

Figure 4-1 Power Supply Outputs

PKW0402A-96
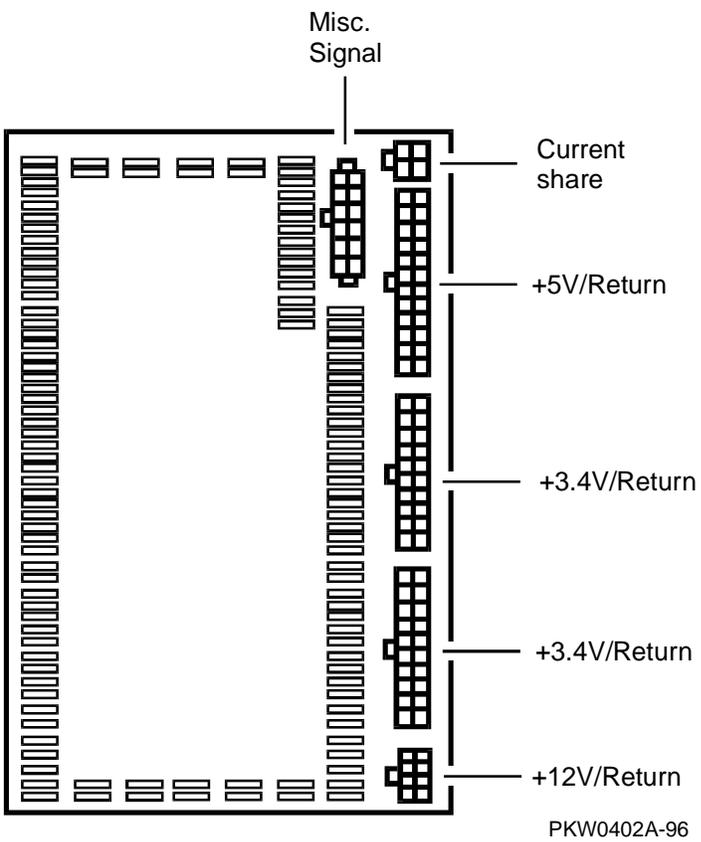
Power Supply Features

- 90–264 Vrms input

- 450 watts output. Output voltages are as follows:

Output Voltage	Min. Voltage	Max. Voltage	Max. Current
+5.0	4.85	5.25	50
+3.43	3.400	3.465	75
+12	11.5	12.6	11
–12	–10.9	–13.2	0.2
–5.0	–4.6	–5.5	0.2
Vaux	8.5	9.5	0.05

- Remote sense on +5.0V and +3.43V

  +5.0V is sensed on all CPUs in the system, the system bus motherboard, and the PCI bus motherboard(s).

  +3.43V is sensed on all CPUs in the system and the system bus motherboard.

- Current share on +5.0V, +3.43V, and +12V.

- 1 % regulation on +3.43V.

- Fault protection (latched). If a fault is detected by the power supply, it will shut down. The faults detected are:

  Overvoltage
  Overcurrent
  Power overload

- DC_ENABLE_L input signal starts the DC outputs.

- POK_H output signal indicates that the power supply is operating properly.

4.2 Power Control Module Features

The power control module (54-24117-01) is located behind the B3040-AA module, the system bus to PCI bus bridge module.

Figure 4-2   Power Control Module

System Motherboard

Power Control
Module Slot

PK-0710-96
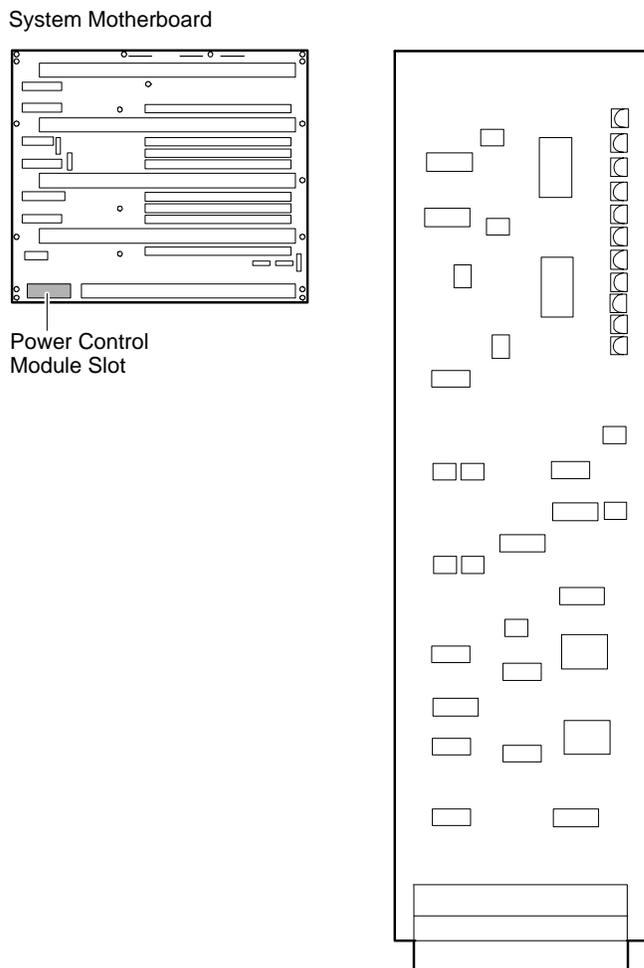
The power control module performs the following functions:

- Controls the power-up/down sequencing.

- Monitors the combined output of power supplies VDD (3.43V) and VCC (5.0V) and asserts DCOK_SENSE if these voltages are within range and asserts POWER_FAULT_L causing an immediate power shutdown if either is not.

- Monitors system temperature and asserts TEMP_FAIL, if temperature exceeds 55° C.

- Monitors CPU and system drawer fans and asserts CPUFAN_OK if all CPU fans are functioning properly, asserts SYSTEM_FAN_OK if the drawer cooling fans are functioning properly; otherwise it asserts FAN_FAULT_L. Each fan is checked at 1 second intervals.

- Powers down the system 30 seconds after detecting TEMP_FAIL, or the absence of CPUFAN_OK, or the absence of SYSTEM_FAN_OK by asserting POWER_FAULT_L.

- Provides visual indication of faults through LEDs.

- Has two registers, one that generates interrupts when bits change, and one that latches errors but does not generate interrupts.

4.3 Power Circuit and Cover Interlocks

Figure 4-3 Power Circuit Diagram
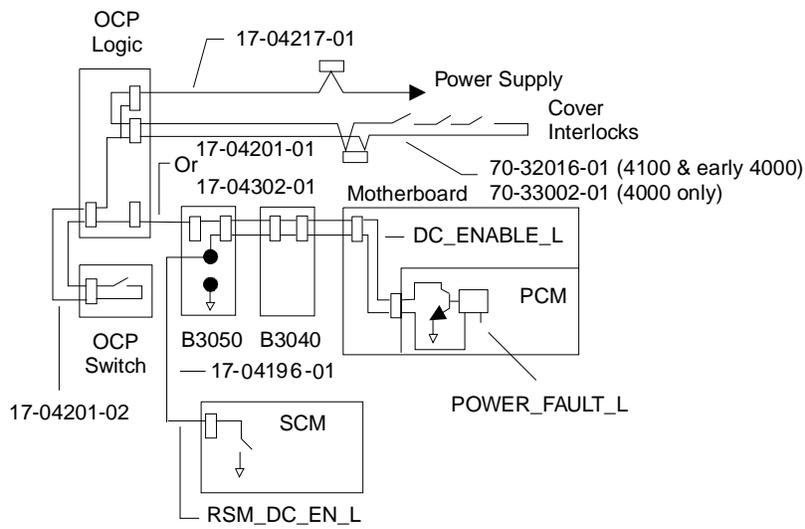
PKW0403F-06

Figure 4-3 shows the distribution of power thoughout the system drawer. Opens in the circuit or the PCM signal POWER_FAULT_L or the SCM signal RSM_DC_EN_L interrupt DC power applied to the system. The opens can be caused by the On/Off button or the cover interlocks. The POWER_FAULT_L signal is asserted by the PCM module if it detects a fault and the RSM_DC_EN_L is controlled remotely.

A failure anywhere in the circuit will result in the removal of DC power. A potential failure is the relay used on the SCM modules to control the RSM_DC_EN_L signal.

The 4100 and early 4000 system drawers have three cover interlocks: one for the system bus card cage, one for the PCI card cage, and one for the power and system fan area. Later 4000 system drawers have four cover interlocks; the fourth switch is for the second PCI card cage.

To override the cover interlocks, find a suitable object to close the interlock circuit at the location identified in Figure 4-3. The switch assembly that contains single switches for all three covers is located where all three covers meet.

4.4 Power-Up/Down Sequence

The On/Off button can be controlled manually or remotely. The button is on the OCP. Remote power control is provided though the remote I/O port connected to the PCI. The power-up/down sequence flow is shown below.

Figure 4-4   Power Up/Down Sequence Flowchart
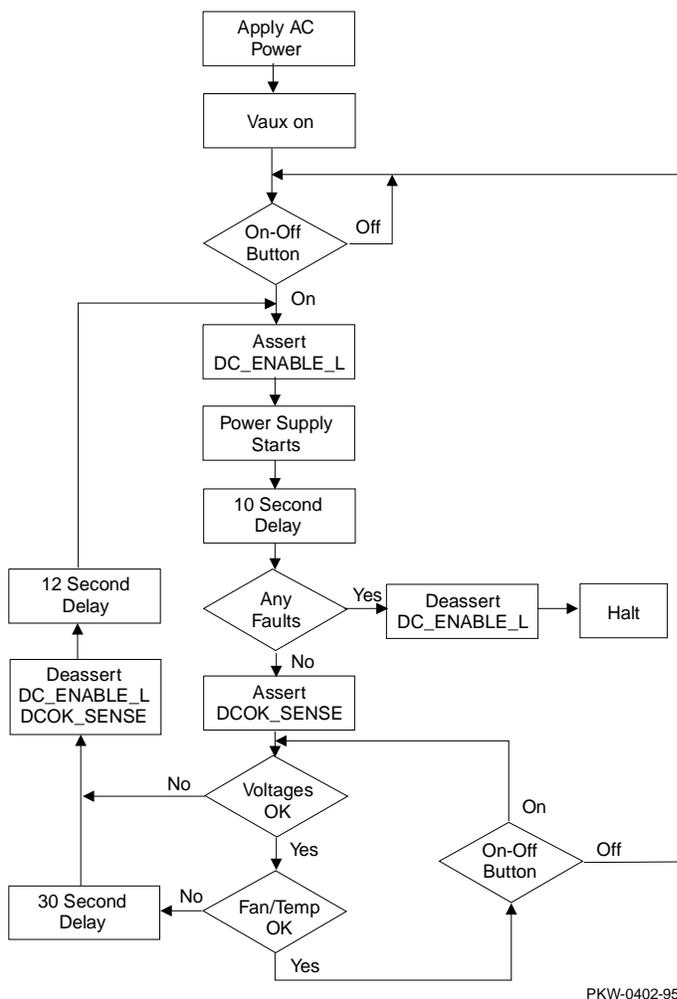
PKW-0402-95

When AC is applied to the system, Vaux (auxiliary voltage) is asserted and is sensed by the PCM. The PCM asserts DC_ENABLE_L starting the power supplies. If there is a hard fault on power-up, the power supplies shut down immediately; otherwise, the power system powers up and remains up until the system is shut off or the PCM senses a fault. If a power fault is sensed, the power system attempts to restore power and will do so if the fault is not sensed a second time. If the fault is still present, the power system shuts down.

Since Vaux is independent of the power supply start, the AC plugs at the front of the supplies must be removed to reset Vaux, allowing capacitors to drain voltage. All power failures require this procedure since the PCM must sense a change in Vaux to start the power supplies.

4.5 Cabinet Power Configuration Rules

There are four cabinets with different power delivery systems. See page 1-9 for a description of differences. A barcode label designating the cabinet variation is located inside the back door in the upper left corner of the bezel holding the door. The four variations are: H9A10-EB, -EC, -EL, -EM.

Figure 4-5   Simple -EB & -EC Cabinet Power Configuration

PKW0406-95
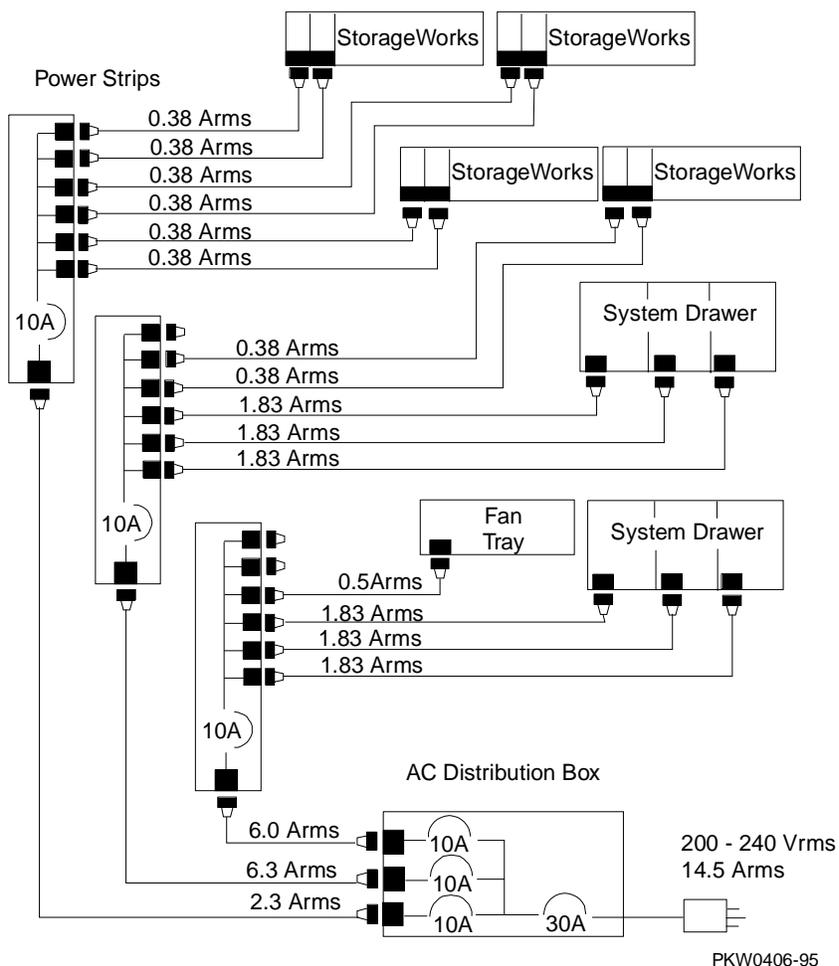
Figure 4-6   Worst-Case -EB & -EC Cabinet Power Configuration

PKW0406A -95
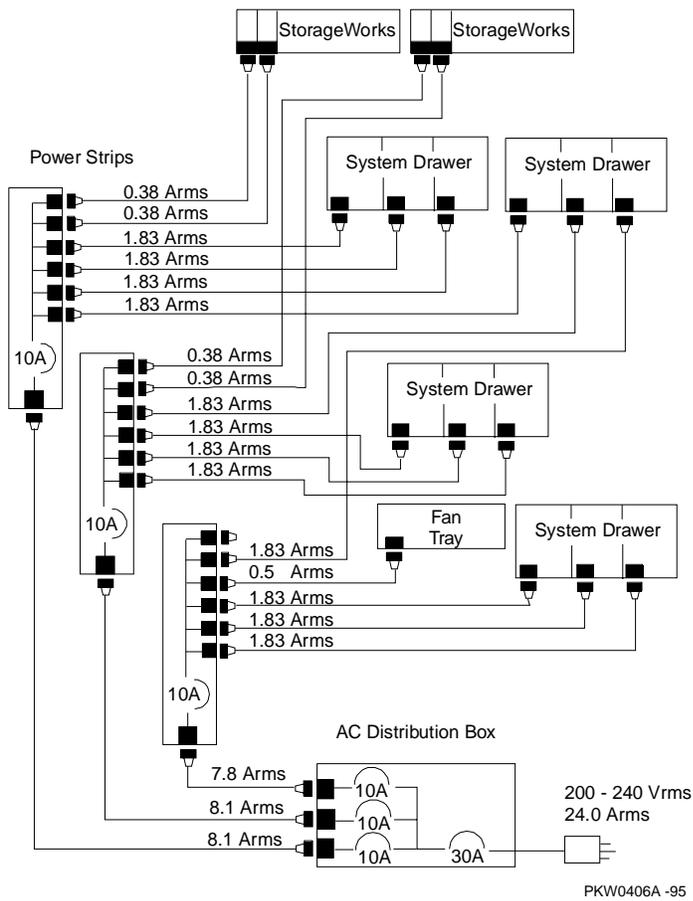
Total Power Available	4800 VA
Single Drawer	1100 VA
Single StorageWorks Shelf	150 VA
System Fan Tray	100 VA
Outlets	18 IEC 320 max. (3 power strips)
Site Grounding	Leakage current exceeds 3.5 mAms.
Power Strip	One system drawer per power strip.
	In four-system drawer configuration, fourth drawer should have its three power cords distributed among the three power strips.

**Figure 4-7   -EL & -EM Single Drawer Cabinet Power Configuration
(Single drawer -EM shown with H7600-DB controller)**

2 Power
Controllers

StorageWorks   StorageWorks

0.38
0.38 A
0.38 Ams   StorageWorks   StorageWorks
0.38 Ams
0.38 Ams
0.38 Ams
0.38 Ams
0.38 Ams
1.83 Ams
1.83 Ams   System Drawer

16 A

Fan tray

0.5
Am   1.83 Ams

StorageWorks   StorageWorks

0.38
0.38 Ams
0.38 Ams   StorageWorks   StorageWorks
0.38 Ams
0.38 Ams
0.38 Ams
0.38 Ams
0.38 Ams

16 A   240 V, 16 AMP Controller with
12 IEC C13 outlets (Europe & A.P.)

240 VMS
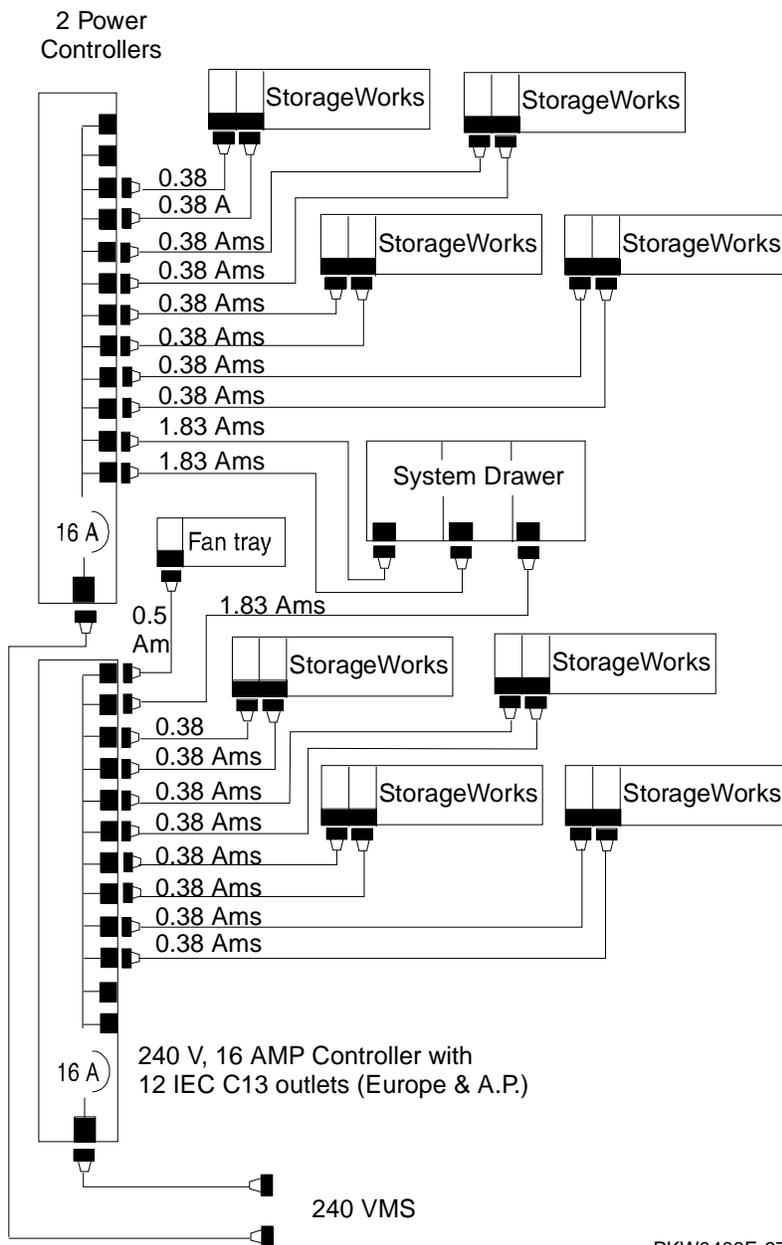
PKW0406E-97

**Figure 4-8   -EL Three Drawer Cabinet Power Configuration
(Three drawer -EL shown with H7600-AA controller)**

2 Power
Controllers

System Drawer

System Drawer

3.67 Ams
3.67 Ams
3.67 Ams
3.67 Ams
3.67 Ams

StorageWorks

15 A

System Drawer

0.75 Ams
0.75 Ams

Fan tray

15 A

1.0 Ams
3.67 Ams
3.67 Ams
3.67 Ams

15 A

StorageWorks

3.67 Ams
0.75 Ams
0.75 Ams

15 A

120 V, 14 AMP Controller with
10 NEMA 5-15 outlets (N.A. & A.P.)

120 VMS
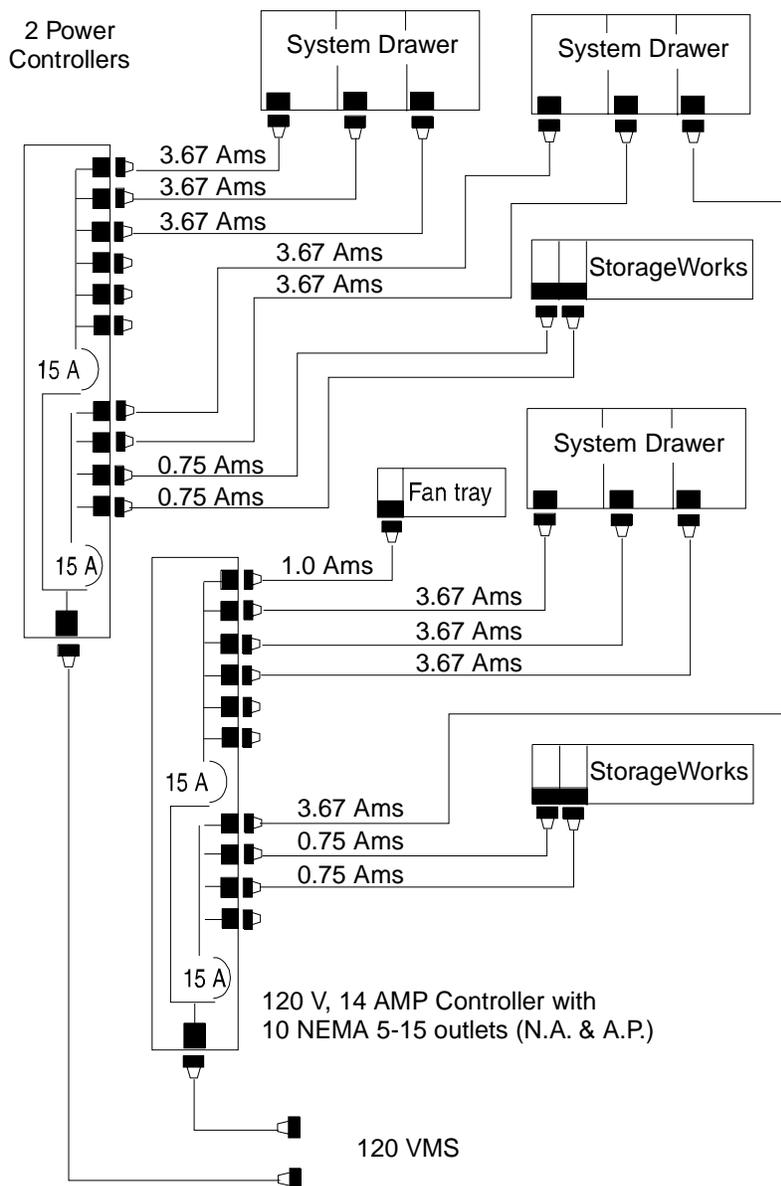
PKW0406F -97

4.6   Pedestal Power Configuration Rules (North America and Japan)

Figure 4-9   Pedestal Power Distribution (N.A. and Japan)
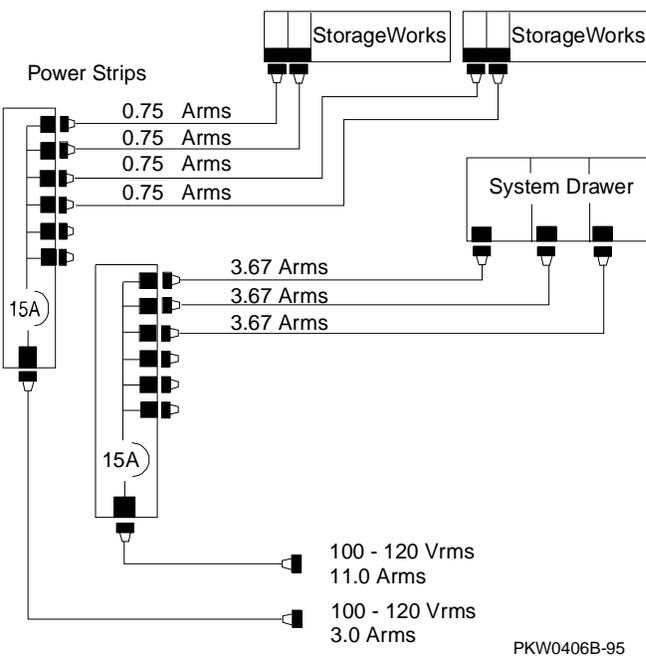
Total Power Available (Assuming a 15 A branch)	N. America: 1800 VA per branch circuit and 1400 VA per line cord Japan: 1500 VA per branch circuit and 1200 VA per line cord
Single Drawer	1100 VA
Single StorageWorks Shelf	150 VA
Outlets	12 NEMA receptacles
Power Strip	Single AC power strip supports one system drawer and one StorageWorks shelf. When two AC power strips are used, combined AC input line current cannot exceed the site circuit breaker restriction, assuming both strips are plugged in to the same circuit.

4.7 Pedestal Power Configuration Rules (Europe and Asia Pacific)

Figure 4-10 Pedestal Power Distribution (Europe and AP)

Power Strips

StorageWorks   StorageWorks

0.34   Arms
0.34   Arms
0.34   Arms
0.34   Arms

System Drawer

10A

1.67 Arms
1.67 Arms
1.67 Arms

10A

200 - 240 Vrms
5.0 Arms
200 - 240 Vrms
3.0 Arms

PKW0406C-95
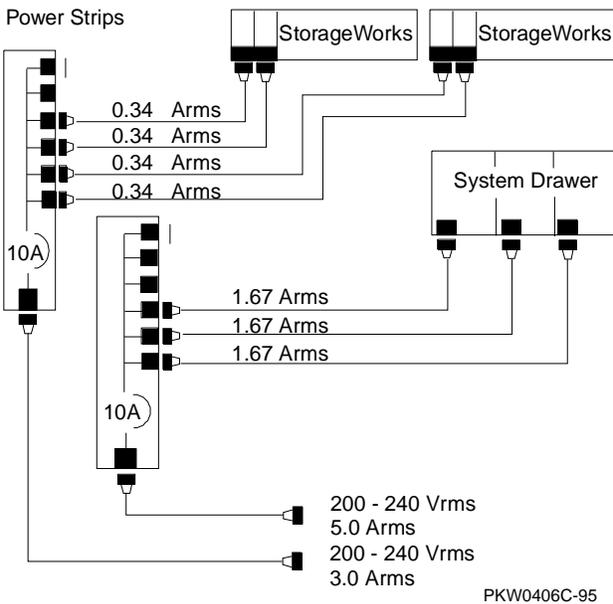
Total Power Available	2200 VA per power strip
Single Drawer	1100 VA
Single StorageWorks Shelf	150 VA
Outlets	10 IEC 320 receptacles max. One receptacle is blocked on each power strip to control leakage.
Power Strip	Single AC power strip supports one system drawer and three StorageWorks shelves.

Chapter 5

Error Logs

This chapter provides information on troubleshooting with error logs. The following topics are covered:

- Using Error Logs
- Using DECevent
- Error Log Examples and Analysis
- Troubleshooting IOD-Detected Errors
- Double Error Halts and Machine Checks While in PAL Mode

Error registers are described in Chapter 6.

5.1 Using Error Logs

Error detection is performed by CPUs, the IOD, and the EISA to PCI bus bridge. (The IOD is the acronym used by software to refer to the system bus to PCI bus bridge.)

Figure 5-1   Error Detector Placement
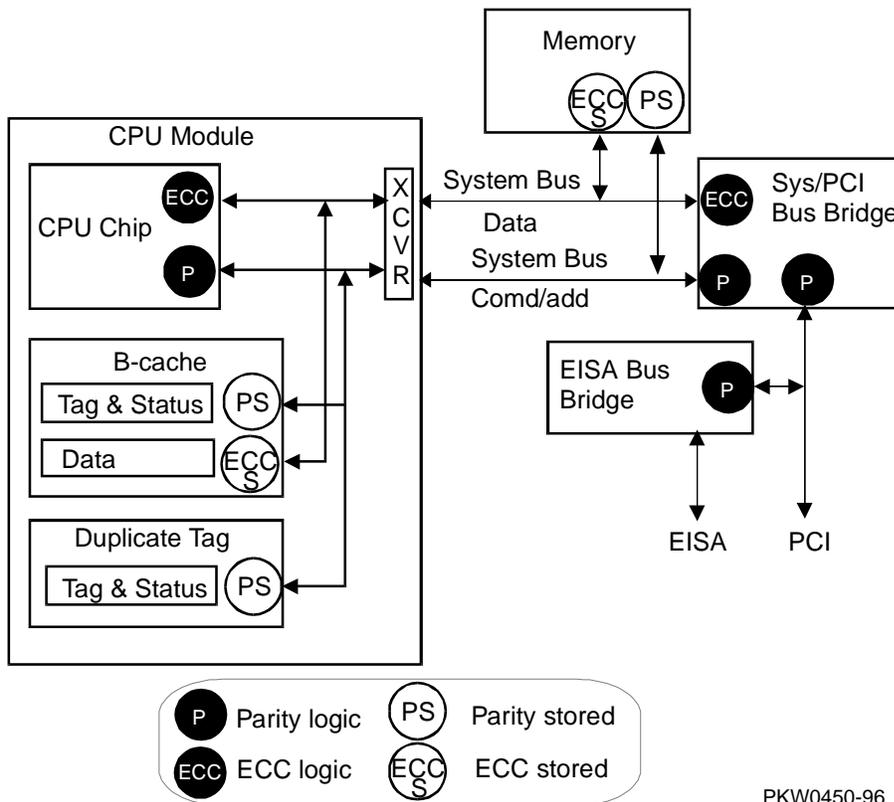
PKW0450-96

Lines Protected	Device
ECC Protected	
System bus data lines	IOD on every transaction, CPU when using the bus
B-cache	IOD on every transaction, CPU when using the bus
Parity Protected	
System bus command/address lines	IOD on every transaction, CPU when using the bus
Duplicate tag store	IOD on every transaction, CPU when using the bus
B-cache index lines	CPU
PCI bus	IOD
EISA bus	EISA bridge

As shown in Figure 5-1 and the accompanying table, the CPU chip is isolated by transceivers (XVER) from the data and command/address lines on the module. This allows the CPU chip access to the duplicate tag and B-cache while the system bus is in use. The CPU detects errors only when it is the consumer of the data. The IOD detects errors on each system bus cycle regardless of whether it is involved in the transaction.

System bus errors detected by the CPU may also be detected by the IOD. It is necessary to check the IOD for errors any time there is a CPU machine check.

- If the CPU sees bad data and the IOD does not, the CPU is at fault.

- If both the CPU and the IOD see bad data on the system bus, either memory or a secondary CPU is the cause. In such a case, the Dirty bit, bit<20>, in the IOD MC_ERR1 Register should be set or clear. If the Dirty bit is set, the source of the data is a CPU's cache destined for a different CPU. If the Dirty bit is not set, memory caused the bad data on the bus. In this case, multiple error log entries occur and must be analyzed together to determine the cause of the error.

5.1.1 Hard Errors

There are two categories of hard errors:

- System-independent errors detected by the CPU. These errors are processor machine checks handled as MCHK 670 interrupts and are:

  Internal EV5 or EV56 cache errors
  CPU B-cache module errors

- System-dependent errors detected by both the CPU and IOD. These errors are system machine checks handled as MCHK 660 interrupts and are:

  CPU-detected external reference errors
  IOD hard error interrupts

The IOD can detect hard errors on either side of the bridge.

5.1.2 Soft Errors

There are two categories of soft errors:

- System-independent errors detected and corrected by the CPU. These errors are CPU module correctable errors handled as MCHK 630 interrupts.

- System-dependent errors that are correctable single-bit errors on the system bus and are handled as MCHK 620 interrupts.

5.1.3 Error Log Events

Several different events are logged by OpenVMS and DIGITAL UNIX. Windows NT does not log errors in this fashion.

Table 5-1   Types of Error Log Events

Error Log Event	Description
MCHK 670	Processor machine checks. These are synchronous errors that inform precisely what happened at the time the error occurred. They are detected inside the CPU chip and are fatal errors.
MCHK 660	System machine checks. These are asynchronous errors that are recorded after the error has occurred. Data on exactly what was going on in the machine at the time of the error may not be known. They are fatal errors.
MCHK 630	Processor correctable errors
MCHK 620	System correctable errors
Last fail	Used to collect system bus registers prior to crashing
I/O error interrupt	IOD error interrupts
System environment	Used to provide status on power, fans, and temperature
Configuration	Used to provide system configuration information

5.2 Using DECevent

DECevent produces bit-to-text ASCII reports derived from system event entries or user-supplied event logs. The format of the reports is determined by commands, qualifiers, parameters, and keywords appended to the comand. The maximum command line length is 255 characters.

DECevent allows you to do the following:

- Translate event log files into readable reports

- Select alternate input and output files

- Filter input events

- Select alternative reports

- Translate events as they occur

- Maintain and customize your environment with the interactive shell commands

To access on-line help:

OpenVMS

```
$ HELP DIAGNOSE  or
$ DIA /INTERACTIVE
DIA> HELP
```

DIGITAL UNIX

```
> man dia   or
> dia hlp
```

Privileges necessary to use DECevent:

- SYSPRV for the utility

- DIAGNOSE to use the /CONTINUOUS qualifier

5.2.1 Translating Event Files

To produce a translated event report using the default event log file,
SYS$ERRORLOG:ERRLOG.SYS, enter the following command:

OpenVMS
```
$ DIAGNOSE
```

DIGITAL UNIX
```
> dia -a
```

The DIAGNOSE command allows DECevent to use built-in defaults. This command
produces a full report, directed to the terminal screen, from the input event file,
SYS$ERRORLOG:ERRLOG.SYS. The /TRANSLATE qualifier is understood on
the command line.

To select an alternate input file

OpenVMS
```
$ DIAGNOSE ERRORLOG.OLD
```

DIGITAL UNIX
```
> dia -a -f syserr-old.hostname
```

These commands select an alternate input file (ERRORLOG.OLD or syserr-old) as
the event log to translate. The file name can contain the directory or path, if needed.
Wildcard characters can be used.

To send reports to an output file

OpenVMS
```
$ DIAGNOSE/OUTPUT=ERRLOG_OLD.TXT
```

DIGITAL UNIX
```
> dia -a > syserr-old.txt
```

These commands direct the output of DECevent to ERRLOG_OLD.TXT or
syserr -old.txt.

To reverse the order of the input events

OpenVMS
```
$ DIAGNOSE/TRANSLATE/REVERSE
```

DIGITAL UNIX
```
> dia -R
```

These commands reverse the order in which events are displayed. The default order
is forward chronologically.

5.2.2 Filtering Events

/INCLUDE and **/EXCLUDE** qualifiers allow you to filter input event log files.

The **/INCLUDE** qualifier is used to create output for devices named in the
command.

OpenVMS
```
$ DIAGNOSE/TRANSLATE/INCLUDE=(DISK=RZ,DISK=RA92,CPU)
```

DIGITAL UNIX
```
> dia -i disk=rz disk=ra92 cpu
```

The commands shown here create output using only the entries for RZ disks, RA92
disks, and CPUs.

The **/EXCLUDE** qualifier is used to create output for all devices except those named
in the command.

OpenVMS
```
$ DIAGNOSE/TRANSLATE/EXCLUDE=(MEMORY)
```

DIGITAL UNIX
```
> dia -x mem
```

Use the **/BEFORE** and **/SINCE** qualifiers to select events before or after a certain date and time.

OpenVMS
```
$ DIAGNOSE/TRANSLATE/BEFORE=15-JAN-1996:10:30:00
```
or
```
$ DIAGNOSE/TRANSLATE/SINCE=15-JAN-1996:10:30:00
```

DIGITAL UNIX
```
> dia -t s:15-jan-1996 e:20-jan-1996
```

If no time is specified, the default time is 00:00:00, and all events for that day are selected.

The **/BEFORE** and **/SINCE** qualifiers can be combined to select a certain period of time.

OpenVMS
```
$ DIAGNOSE/TRANSLATE/SINCE=15-JAN-1996/BEFORE=20-JAN-1996
```

If no value is supplied with the **/SINCE** or **/BEFORE** qualifiers, DECevent defaults to **TODAY**.

5.2.3 Selecting Alternative Reports

Table 5-2 describes the DECevent report formats. Report formats are mutually exclusive. No combinations are allowed. The default format is /Full.

Table 5-2   DECevent Report Formats

Format	Description
/Full	Translates all available information for each event
/Brief	Translates key information for each event
/Terse	Provides binary event information and displays register values and other ASCII messages in a condensed format
/Summary	Produces a statistical summary of the events in the log
/Fsterr	Produces a one-line-per-entry report for disk and tape devices

The syntax is:

OpenVMS
```
$ DIAGNOSE/TRANSLATE/<format>
```

DIGITAL UNIX
```
> dia -o <format>
```

5.3 Error Log Examples and Analysis

The following sections provide examples and analysis of error logs.

5.3.1 MCHK 670 CPU-Detected Failure

The error log in Example 5-1 shows the following:

❶   CPU1 logged the error in a system with two CPUs.

❷   During a D-ref fill, the External Interface Status Register logged an uncorrectable EEC error. (When a CPU chip does not find data it needs to perform a task in any of its caches, it requests data from off the chip to fill its D-caches. It performs a "D-ref fill.") Bit<30> is clear, indicating that the source of the error is the B-cache.

❸   Neither IOD CAP Error Register saw an error.

The error was detected by a CPU and the data was not on the system bus. Otherwise, the IODs would have seen the error. Therefore, CPU1 is broken.

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Example 5-1   MCHK 670

```
Logging OS                      2. DIGITAL UNIX
System Architecture             2. Alpha
Event sequence number           4.
Timestamp of occurrence            04-APR-1996 17:20:04
Host name                          whip16

System type register    x00000016  AlphaStation 4x00
Number of CPUs (mpnum)   x00000002                        ❶
CPU logging event (mperr) x00000001


Event validity                  1. O/S claims event is valid
Event severity                  1. Severe Priority
Entry type                    100. CPU Machine Check Errors

CPU Minor class                 1. Machine check (670 entry)

Software Flags         x0000000300000000
                                   IOD 1 Register Subpkt Pres
                                   IOD 2 Register Subpkt Pres
Active CPUs             x00000003
Hardware Rev           x00000000
System Serial Number               C1563
Module Serial Number
Module Type               x0000
System Revision        x00000000

* MCHK 670 Regs *
Flags:                 x00000000
PCI Mask                  x0000
Machine Check Reason      x0098
PAL SHADOW REG 0       x00000000
PAL SHADOW REG 1       x00000000
.
.
.
PAL SHADOW REG 6       x00000000
PAL SHADOW REG 7       x00000000
PALTEMP0               x00000000E87C7A58
PALTEMP1               xFFFFFFFE8F658000
PALTEMP2               xFFFFFC00003C9F40
.
.
.
PALTEMP22              xFFFFFC00004F9D60
PALTEMP23              x00000000E8709A58
Exception Address Reg  xFFFFFC00003BFB88
                                   Native-mode instruction
                                   Exception PC   x3FFFFF00000EFEE2
Exception Summary Reg  x00000000
Exception Mask Reg     x00000000
PAL BASE               x000000000020000
                                   Base addr for palcode =
x0000000008
Interrupt Summary Reg  x00000000
                                   AST requests 3 - 0   x00000000
IBOX Ctrl and Status Reg x000000C160000000
                                   Timeout Bit Not Set
                                   PAL Shadow Registers Enabled
                                   Correctable Err Intrpts Enabled
                                   ICACHE BIST Successful
```

```
                                         TEST_STATUS_H Pin Asserted
Icache Par Err Stat Reg   x00000000
Dcache Par Err Stat Reg   x00000000
Virtual Address Reg       xFFFFFFFE8F63BD38
Memory Mgmt Flt Sts Reg   x000000000166D1
                                         Ref which caused err was a write
                                         Ref resulted in DTB miss
                                         RA Field   x0000000000001B
                                         Opcode Field    x0000000000002C
Scache Address Reg        xFFFFFF00000254BF
Scache Status Reg         x00000000
Bcache Tag Address Reg    xFFFFFF80E98F7FFF
                                         External cache hit
                                         Parity for ds and v bits
                                         Cache block dirty
                                         Cache block valid
                                         Ext cache tag addr parity bit
                                         Tag address<38:20> is
x00000000000E98
Ext Interface Address Reg xFFFFFF00E984DBCF
Fill Syndrome Reg         x0000000000002B
```

Ext Interface Status Reg xFFFFFFF104FFFFFF   ❷
                           Uncorrectable ECC error
                           Error occurred during D-ref fill

```
LD LOCK                   xFFFFFF003797340F

** IOD SUBPACKET -> **                    IOD 0 Register Subpacket
WHOAMI                    x000000BB  Device ID  x0000003B
                                     Bcache Size = 2MB
                                     VCTY ASIC Rev = 0
                                     Module Revision  0.
Base Address of Bridge    x000000F9E0000000
PCI Revision              x06008021  CAP Chip Revision  x00000001
                                     Horse Module Revision  x00000002
                                     Saddle Module Revision  x00000000
                                     Saddle Module Type  Left Hand
                                     EISA Present
                                     PCI Class Code  x00000600
MC-PCI Command Register   x06480FF1  Selftest passed
                                     Delayed read enabled
                                     Bridge PCI trans enabled
                                     Req 64 bit data trans enabled
                                     Accept 64 bit data trans enabled
                                     Check PCI Addr Parity enabled
                                     Check MC bus CMS/Addr Parity
                                     enabled
                                     Check MC bus NXM enabled
                                     Check all transaction enabled
                                     16 byte aligned block write enabled
                                     Write Pend Number Thresho x00000008
                                     RD_TYPE   Short
                                     RL_TYPE   Medium
                                     RM_TYPE   Long
                                     ARB_MODE  MC-PCI Bridge Priority
                                     Mode
Memory Host Addr Exten    x00000000
IO Host Addr Extension    x00000000
Interrupt Control         x00000003  MC-PCI Intr Enabled
                                     Device intr info enabled if en_int=
                                     1
Interrupt Request         x00000000  Interrupts asserted  x00000000
Interrupt Mask Register 0 x00C50010
Interrupt Mask Register 1 x00000000
MC Error Info Register 0  xE0000000  MC bus trans addr <31:4>  x0E000000
MC Error Info Register 1  x000E88FD  MC bus trans addr <39:32>x000000FD
```

Error Logs **5-13**

```
                                    MC_Command x00000008
                                    Device Id x0000003A
CAP Error Register          x00000000       (no error seen)             ❸
PCI Bus Trans Error Adr     x00000000
MDPA Status Register        x00000000  MDPA Chip Revision   x00000000
MDPA Error Syndrome Reg     x00000000  Cycle 0 ECC Syndrome x00000000
                                       Cycle 1 ECC Syndrome x00000000
                                       Cycle 2 ECC Syndrome x00000000
                                       Cycle 3 ECC Syndrome x00000000
MDPB Status Register        x00000000  MDPB Chip Revision   x00000000
MDPB Error Syndrome Reg     x00000000  Cycle 0 ECC Syndrome x00000000
                                       Cycle 1 ECC Syndrome x00000000
                                       Cycle 2 ECC Syndrome x00000000
                                       Cycle 3 ECC Syndrome x00000000


** IOD SUBPACKET -> **                 IOD 1 Register Subpacket
WHOAMI                      x000000BB  Device ID   x0000003B
                                       Bcache Size = 2MB
                                       VCTY ASIC Rev = 0
                                       Module Revision   0.
Base Address of Bridge      x000000FBE0000000
PCI Revision                x06000021  CAP Chip Revision   x00000001
                                       Horse ModuleRevision   x00000002
                                       Saddle Module Revision   x00000000
                                       Saddle Module Type   Left Hand
                                       PCI Class Code   x00000600
MC-PCI Command Register     x06480FF1  Selftest passed
                                       Delayed read enabled
                                       Bridge PCI trans enabled
                                       Req 64 bit data trans enabled
                                       Accept 64 bit data trans enabled
                                       Check PCI Addr Parity enabled
                                       Check MC bus CMS/Addr Parity
                                       enabled
                                       Check MC bus NXM enabled
                                       Check all transaction enabled
                                       16 byte aligned block write enabled
                                       Write Pend Number Thresho x00000008
                                       RD_TYPE   Short
                                       RL_TYPE   Medium
                                       RM_TYPE   Long
                                       ARB_MODE   MC-PCI Bridge Priority
                                       Mode
Memory Host Addr Exten      x00000000
IO Host Addr Extension      x00000000
Interrupt Control           x00000003  MC-PCI Intr Enabled
                                       Device intr info enabled if en_int
                                       = 1
Interrupt Request           x00000000  Interrupts asserted  x00000000
Interrupt Mask Register 0 x00C50001
Interrupt Mask Register 1 x00000000
MC Error Info Register 0  xE0000000  MC bus trans addr <31:4>  x0E000000
MC Error Info Register 1  x000E88FD  MC bus trans addr <39:32> x000000FD
                                       MC_Command x00000008
                                       Device Id x0000003A
CAP Error Register          x00000000       (no error seen)             ❸
PCI Bus Trans Error Adr     xC0018B48
MDPA Status Register        x00000000  MDPA Chip Revision   x00000000
MDPA Error Syndrome Reg     x00000000  Cycle 0 ECC Syndrome x00000000
                                       Cycle 1 ECC Syndrome x00000000
                                       Cycle 2 ECC Syndrome x00000000
                                       Cycle 3 ECC Syndrome x00000000
MDPB Status Register        x00000000  MDPB Chip Revision   x00000000
MDPB Error Syndrome Reg     x00000000  Cycle 0 ECC Syndrome x00000000
                                       Cycle 1 ECC Syndrome x00000000
```

```
                              Cycle 2 ECC Syndrome x00000000
                              Cycle 3 ECC Syndrome x00000000


PALcode Revision              Palcode Rev: 1.21-3
```

5.3.2 MCHK 670 CPU and IOD-Detected Failure

The error log in Example 5-2 shows the following:

❶   CPU3 logged the error in a system with four CPUs.

❷   The External Interface Status Register logged an uncorrectable ECC error during a D-ref fill. (When a CPU chip does not find data it needs to perform a task in any of its caches, it requests data from off the chip to fill its D-cache. It performs a "D-ref fill.") Bit <30> is set, indicating that the source of the error is memory or the system. Bits <32> and <35> are set, indicating an uncorrectable ECC error and a second external interface hard error, respectively.

❸   Both IOD CAP Error Registers logged an error.

❹   The command at the time of the error was a read.

❺   The bus master at the time of the error was CPU3.

❻   The Dirty bit, bit <20> in the MC_ERR1 Register is clear, indicating the data is clean and comes from memory.

The error was detected by a CPU, and the data was on the system bus and is clean. Therefore, a memory module provided the wrong data. (If the Dirty bit had been set, the data would have come from the cache of another CPU.) To determine which memory, see Section 5.4

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Example 5-2 MCHK 670 CPU and IOD-Detected Failure

```
Logging OS                        2. DIGITAL UNIX
System Architecture               2. Alpha
Event sequence number             6.
Timestamp of occurrence              08-APR-1996 11:27:55
Host name                            whip16

System type register      x00000016  AlphaStation 4x00
```

Number of CPUs (mpnum)   x00000004   ❶
CPU logging event (mperr) x00000003

```
Event validity                    1. O/S claims event is valid
Event severity                    1. Severe Priority
Entry type                      100. CPU Machine Check Errors

CPU Minor class                   1. Machine check (670 entry)

Software Flags            x0000000300000000
                                     IOD 1 Register Subpkt Pres
                                     IOD 2 Register Subpkt Pres
Active CPUs               x0000000F
Hardware Rev             x00000000
System Serial Number                 C1563
Module Serial Number
Module Type                  x0000
System Revision          x00000000

* MCHK 670 Regs *
Flags:                   x00000000
PCI Mask                     x0000
Machine Check Reason         x0098
PAL SHADOW REG 0         x00000000
PAL SHADOW REG 1         x00000000
.
.
.
PAL SHADOW REG 6         x00000000
PAL SHADOW REG 7         x00000000
PALTEMP0                 x00000001401A7A90
PALTEMP1                 x00000000000021
.
.
.
PALTEMP23                x00000000ECE77A58
Exception Address Reg    x000000012005A8B4
                                  Native-mode instruction
                                  Exception PC  x0000000048016A2D
Exception Summary Reg    x00000000
Exception Mask Reg       x00000000
PAL BASE                 x000000000020000
                                  Base addr for palcode =
x0000000008
Interrupt Summary Reg    x00000000
                                  AST requests 3 - 0  x00000000
IBOX Ctrl and Status Reg x000000C164000000
                                  Timeout Bit Not Set
                                  Floating Point Instr. may be issued
                                  PAL Shadow Registers Enabled
                                  Correctable Err Intrpts Enabled
                                  ICACHE BIST Successful
                                  TEST_STATUS_H Pin Asserted
Icache Par Err Stat Reg  x00000000
```

```
Dcache Par Err Stat Reg   x00000000
Virtual Address Reg       x00000001407D6000
Memory Mgmt Flt Sts Reg   x00000000011A10
                                      Ref resulted in DTB miss
                                      RA Field   x0000000008

                                      Opcode Field    x00000000000023
Scache Address Reg        xFFFFFF00000254BF
Scache Status Reg         x00000000
Bcache Tag Address Reg    xFFFFFF80286F7FFF
                                      External cache hit
                                      Parity for ds and v bits
                                      Cache block dirty
                                      Cache block valid
                                      Ext cache tag addr parity bit
                                      Tag address<38:20> is
x00000000000286
Ext Interface Address Reg xFFFFFF0028681A8F
Fill Syndrome Reg         x00000000004B00

Ext Interface Status Reg  xFFFFFFFF984FFFFFF                        ❷
                                      Uncorrectable ECC error
                                      Error occurred during D-ref fill
                                      Second external interface hard
                                      error
LD LOCK                   xFFFFFF000020040F


** IOD SUBPACKET -> **            IOD 0 Register Subpacket
WHOAMI                    x000000BF Device ID  x0000003F
                                    Bcache Size = 2MB
                                    VCTY ASIC Rev = 0
                                    Module Revision  0.
Base Address of Bridge    x000000F9E0000000
PCI Revision              x06008021 CAP Chip Revision  x00000001
                                    Horse Module Revision  x00000002
                                    Saddle Module Revision  x00000000
                                    Saddle Module Type  Left Hand
                                    EISA Present
                                    PCI Class Code  x00000600
MC-PCI Command Register   x06460FF1 Selftest passed
                                    Delayed read enabled
                                    Bridge PCI trans enabled
                                    Req 64 bit data trans enabled
                                    Accept 64 bit data trans enabled
                                    Check PCI Addr Parity enabled
                                    Check MC bus CMS/Addr Parity
                                    enabled
                                    Check MC bus NXM enabled
                                    Check all transaction enabled
                                    16 byte aligned block write enabled
                                    Write Pend Number Thresho x00000006
                                    RD_TYPE   Short
                                    RL_TYPE   Medium
                                    RM_TYPE   Long
                                    ARB_MODE  MC-PCI Bridge Priority
                                    Mode
Memory Host Addr Exten    x00000000
IO Host Addr Extension    x00000000
Interrupt Control         x00000003 MC-PCI Intr Enabled
                                    Device intr info enabled if en_int
                                    = 1
Interrupt Request         x00810000 Interrupts asserted  x00010000
                                    Hard Error
Interrupt Mask Register 0 x00C50010
Interrupt Mask Register 1 x00000000
```

```
MC Error Info Register 0  x28681A80  MC bus trans addr <31:4>  x028681A8   ❻
MC Error Info Register 1  x800FD800  MC bus trans addr <39:32> x00000000

                                     MC_Command x00000018                   ❹

                                     Device Id x0000003F                    ❺
                                     MC error info valid
CAP Error Register         xC0000000  Uncorrectable ECC err det by MDPB

                                     MC error info latched                 ❸
PCI Bus Trans Error Adr    x000003FD
MDPA Status Register       x00000000  MDPA Chip Revision  x00000000
MDPA Error Syndrome Reg    x00000000  Cycle 0 ECC Syndrome x00000000
                                      Cycle 1 ECC Syndrome x00000000
                                      Cycle 2 ECC Syndrome x00000000
                                      Cycle 3 ECC Syndrome x00000000
MDPB Status Register       x80000000  MDPB Chip Revision  x00000000
                                      MPDB Error Syndrome of
                                      uncorrectable read error

MDPB Error Syndrome Reg    x0000004B  Cycle 0 ECC Syndrome
x0000000000004B
                                      Cycle 1 ECC Syndrome x00000000
                                      Cycle 2 ECC Syndrome x00000000
                                      Cycle 3 ECC Syndrome x00000000


** IOD SUBPACKET -> **                IOD 1 Register Subpacket
WHOAMI                     x000000BF  Device ID  x0000003F
                                      Bcache Size = 2MB
                                      VCTY ASIC Rev = 0
                                      Module Revision  0.
Base Address of Bridge     x000000FBE0000000
PCI Revision               x06000021  CAP Chip Revision  x00000001
                                      Horse Module Revision  x00000002
                                      Saddle Module Revision  x00000000
                                      Saddle Module Type  Left Hand
                                      PCI Class Code  x00000600
MC-PCI Command Register    x06460FF1  Selftest passed
                                      Delayed read enabled
                                      Bridge PCI trans enabled
                                      Req 64 bit data trans enabled
                                      Accept 64 bit data trans enabled
                                      Check PCI Addr Parity enabled
                                      Check MC bus CMS/Addr Parity
                                      enabled
                                      Check MC bus NXM enabled
                                      Check all transaction enabled
                                      16 byte aligned block write enabled
                                      Write Pend Number Thresho x00000006
                                      RD_TYPE   Short
                                      RL_TYPE   Medium
                                      RM_TYPE   Long
                                      ARB_MODE  MC-PCI Bridge Priority
                                      Mode
Memory Host Addr Exten     x00000000
IO Host Addr Extension     x00000000
Interrupt Control          x00000003  MC-PCI Intr Enabled
                                      Device intr info enabled if en_int
                                      = 1
Interrupt Request          x00800000  Interrupts asserted  x00000000
                                      Hard Error
Interrupt Mask Register 0 x00C50001
Interrupt Mask Register 1 x00000000

MC Error Info Register 0  x28681A80  MC bus trans addr <31:4>  x028681A8   ❻
MC Error Info Register 1  x800FD800  MC bus trans addr <39:32> x00000000
```

```
                                  MC_Command x00000018              ❹

                                  Device Id x0000003F               ❺
                                  MC error info valid
CAP Error Register       xC0000000  Uncorrectable ECC err det by MDPB

                                  MC error info latched            ❸
PCI Bus Trans Error Adr   x00000000
MDPA Status Register      x00000000  MDPA Chip Revision   x00000000
MDPA Error Syndrome Reg   x00000000  Cycle 0 ECC Syndrome x00000000
                                     Cycle 1 ECC Syndrome x00000000
                                     Cycle 2 ECC Syndrome x00000000
                                     Cycle 3 ECC Syndrome x00000000
MDPB Status Register      x80000000  MDPB Chip Revision   x00000000
                                     MPDB Error Syndrome of
                                     uncorrectable read error
MDPB Error Syndrome Reg   x0000004B  Cycle 0 ECC Syndrome
x0000000000004B
                                     Cycle 1 ECC Syndrome x00000000
                                     Cycle 2 ECC Syndrome x00000000
                                     Cycle 3 ECC Syndrome x00000000

PALcode Revision                     Palcode Rev: 1.21-3
```

5.3.3 MCHK 670 Read Dirty CPU-Detected Failure

The error log in Example 5-3 shows the following:

❶   CPU0 logged the error in a system with two CPUs.

❷   The External Interface Status Register records an uncorrectable ECC error from the system (bit <30> set).

❸   Both IOD CAP Error Registers logged an error.

❹   The MC Error Info Registers 0 and 1 have captured the error information.

❺   The commander at the time of the error was CPU0 (known from MC_ERR1)

❻   The command on the bus at the time was a read memory command.

❼   The address read was a memory address, not an I/O address.

❽   The data associated with the read was dirty.

From this information you know CPU0 requested data that was dirty; therefore, memory did not provide it, nor did an I/O device. Only another CPU could have provided the data from its cache. There is only one other CPU in this system, and it is faulty. Had there been more than two CPUs you could not have identified the error to a particular CPU. See Section 5.4 for a procedure designed to help with IOD-detected errors.

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Example 5-3   MCHK 670 Read Dirty Failure

```
Logging OS                        2. DIGITAL UNIX
System Architecture               2. Alpha
Event sequence number             4.
Timestamp of occurrence              08-APR-1996 10:20:37
Host name                            sect06

System type register     x00000016  AlphaStation 4x00
Number of CPUs (mpnum)   x00000002

CPU logging event (mperr) x00000000                        ❶


Event validity                    1. O/S claims event is valid
Event severity                    1. Severe Priority
Entry type                      100. CPU Machine Check Errors

CPU Minor class                   1. Machine check (670 entry)

Software Flags           x0000000300000000
                                      IOD 0 Register Subpkt Pres
                                      IOD 1 Register Subpkt Pres
Active CPUs               x00000003
Hardware Rev             x00000000
System Serial Number                 C1563
Module Serial Number
Module Type                 x0000
System Revision          x00000000

* MCHK 670 Regs *
Flags:                   x00000000
PCI Mask                    x0000
Machine Check Reason        x0098  Fatal Alpha Chip Detected HardError
PAL SHADOW REG 0         x0000000000000000
PAL SHADOW REG 1         x0000000000000000
PAL SHADOW REG 2         x0000000000000000
PAL SHADOW REG 3         x0000000000000000
PAL SHADOW REG 4         x0000000000000000
PAL SHADOW REG 5         x0000000000000000
PAL SHADOW REG 6         x0000000000000000
PAL SHADOW REG 7         x0000000000000000
PALTEMP0                 xFFFFFC00006C00C0
PALTEMP1                 x00000000000061A8
PALTEMP2                 xFFFFFC00004E1E00
.
.
.
PALTEMP22                xFFFFFC00006530E0
PALTEMP23                x0000000003D2BA58
Exception Address Reg    xFFFFFC000047395C
                                      Native-mode Instruction
                                      Exception PC  x3FFFFF000011CE57
Exception Summary Reg    x0000000000000000
Exception Mask Reg       x0000000000000000
PAL Base Address Reg     x0000000000020000
                                      Base Addr for PALcode:
x0000000000000008
Interrupt Summary Reg    x0000000000200000
                                      External HW Interrupt at IPL21
                                      AST Requests 3-0:
x0000000000000000
IBOX Ctrl and Status Reg x000000C160000000
                                      Timeout Counter Bit Clear.
                                      IBOX Timeout Counter Enabled.
```

```
                                    Floating Point Instructions will
                                    cause FEN Exceptions.
                                    PAL Shadow Registers Enabled.
                                    Correctable Error Interrupts
                                    Enabled.
                                    ICACHE BIST (Self Test) Was
                                    Successful.
                                    TEST_STATUS_H Pin Asserted
Icache Par Err Stat Reg    x0000000000000000
Dcache Par Err Stat Reg    x0000000000000000
Virtual Address Reg        x0000000000044000
Memory Mgmt Flt Sts Reg    x0000000000005D10
                                    If Err, Reference Resulted in DTB
                                    Miss
                                    Fault Inst RA Field:
x0000000000000014

                                    Fault Inst Opcode:
x000000000000000B
Scache Address Reg         xFFFFFF00000254BF
Scache Status Reg          x0000000000000000
Bcache Tag Address Reg     xFFFFFF8007EE2FFF
                                    Last Bcache Access Resulted in a
                                    Miss.
                                    Value of Parity Bit for Tag Control
                                    Status
                                     Bits Dirty, Shared & Valid is Set.
                                    Value of Tag Control Dirty Bit is
                                    Clear.
                                    Value of Tag Control Shared Bit is
                                    Clear.
                                    Value of Tag Control Valid Bit is
                                    Clear.
                                    Value of Parity Bit Covering Tag
                                    Store ddress Bits is Set.
                                    Tag Address<38:20> Is:
x000000000000007E
```
Ext Interface Address Reg xFFFFFF0007FBF08F
Fill Syndrome Reg   x000000000000D189

Ext Interface Status Reg   xFFFFFFFF944FFFFFF   ❷
                                    Error Source is Memory or System
                                    UNCORRECTABLE ECC ERROR
                                    Error Occurred During D-ref Fill
                                    Error
```
LD LOCK                    xFFFFFF0007FBF00F

** IOD SUBPACKET -> **               IOD 0 Register Subpacket

WHOAMI                    x000000BA  Module Revision  0.
                                    VCTY ASIC Rev = 0
                                    Bcache Size = 2MB
                                    MID  2.
                                    GID  7.

Base Address of Bridge    x000000F9E0000000
Dev Type & Rev Register   x06008021  CAP Chip Revision:       x00000001
                                    HORSE  Module Revision:   x00000002
                                    SADDLE Module Revision:   x00000000
                                    SADDLE Module Type:       LeftHand
                                    PCI-EISA Bus Bridge Present on PCI
                                    Segment
                                    PCI Class Code            x00000600
MC-PCI Command Register   x06480FF1  Module SelfTest Passed LED on
                                    Delayed PCI Bus Reads Protocol:
                                    Enabled
```

```
                                        Bridge to PCI Transactions: Enabled
                                        Bridge REQUESTS 64 Bit Data
                                        Transactions
                                        Bridge ACCEPTS 64 Bit Data
                                        Transactions
                                        PCI Address Parity Check: Enabled
                                        MC Bus CMD/Addr Parity Check:
                                        Enabled
                                        MC Bus NXM Check: Enabled
                                        Check ALL Transactions for Errors
                                        Use MC_BMSK for 16 Byte Align Blk
                                        Mem Wrt
                                        Wrt PEND_NUM Threshold:  8.
                                        RD_TYPE Memory Prefetch Algorithm:
                                        Short
                                        RL_TYPE Mem Rd Line Prefetch Type:
                                        Medium
                                        RM_TYPE Mem Rd Multiple Cmd Type:
                                        Long
                                        ARB_MODE Arbitration: MC-PCI
Priority Mode
Mem Host Address Ext Reg  x00000000   HAE Sparse Mem Adr<31:27> x00000000
IO Host Adr Ext Register  x00000000   PCI Upper Adr Bits<31:25> x00000000
Interrupt Ctrl Register   x00000003   Write Device Interrupt Info
Struct:Enabled
Interrupt Request         x00800000   Interrupts asserted  x00000000
                                      Hard Error

Interrupt Mask0 Register  x00C50010
Interrupt Mask1 Register  x00000000
MC Error Info Register 0  x07FBF080

                                      MC Bus Trans Addr<31:4>: 7FBF080

MC Error Info Register 1  x801E8800   MC bus trans addr <39:32> x00000000    ❼

                                      MC Command is Read0-Mem                 ❻

                                      Device ID 2  x00000002                  ❺

                                      MC bus error assoc w read/dirty         ❽
                                      MC error info valid

CAP Error Register        xE0000000   Uncorrectable ECC err det by MDPA       ❸
                                      Uncorrectable ECC err det by MDPB

                                      MC error info latched                   ❹

Sys Environmental Regs    x00000000
PCI Bus Trans Error Adr   x00000000
MDPA Status Register      xC0000000   MDPA Status Register Data Not Valid
MDPA Error Syndrome Reg   x00080089   MDPA Syndrome Register Data Not
Valid
MDPB Status Register      x80000000   MDPB Status Register Data Not Valid
MDPB Error Syndrome Reg   x000D00D1   MDPB Syndrome Register Data Not
Valid

** IOD SUBPACKET -> **                IOD 1 Register Subpacket

WHOAMI                    x000000BA   Module Revision  0.
                                      VCTY ASIC Rev = 0
                                      Bcache Size = 2MB
                                      MID  2.
                                      GID  7.

Base Address of Bridge    x000000FBE0000000
Dev Type & Rev Register   x06000021   CAP Chip Revision:        x00000001
                                      HORSE  Module Revision:   x00000002
                                      SADDLE Module Revision:   x00000000
                                      SADDLE Module Type:       LeftHand
                                      Internal CAP Chip Arbiter: Enabled
```

```
                                      PCI Class Code           x00000600
MC-PCI Command Register   x06480FF1   Module SelfTest Passed LED on
                                      Delayed PCI Bus Reads Protocol:
                                      Enabled
                                      Bridge to PCI Transactions: Enabled
                                      Bridge REQUESTS 64 Bit Data
                                      Transactions
                                      Bridge ACCEPTS 64 Bit Data
                                      Transactions
                                      PCI Address Parity Check: Enabled
                                      MC Bus CMD/Addr Parity Check:
                                      Enabled
                                      MC Bus NXM Check: Enabled
                                      Check ALL Transactions for Errors
                                      Use MC_BMSK for 16 Byte Align Blk
Mem Wrt
                                      Wrt PEND_NUM Threshold:  8.
                                      RD_TYPE Memory Prefetch Algorithm:
                                      Short
                                      RL_TYPE Mem Rd Line Prefetch Type:
                                      Medium
                                      RM_TYPE Mem Rd Multiple Cmd Type:
                                      Long
                                      ARB_MODE Arbitration: MC-PCI
                                      Priority Mode
Mem Host Address Ext Reg  x00000000   HAE Sparse Mem Adr<31:27> x00000000
IO Host Adr Ext Register  x00000000   PCI Upper Adr Bits<31:25> x00000000
Interrupt Ctrl Register   x00000003   Write Device Interrupt Info
Struct:Enabled
Interrupt Request         x00800001   Interrupts asserted  x00000001
                                      Hard Error

Interrupt Mask0 Register  x00C50001
Interrupt Mask1 Register  x00000000
MC Error Info Register 0  x07FBF080

                                      MC Bus Trans Addr<31:4>: 7FBF080  ❼
MC Error Info Register 1  x801E8800   MC bus trans addr <39:32> x00000000

                                      MC Command is Read0-Mem           ❻

                                      Device ID 2  x00000002            ❺

                                      MC bus error assoc w read/dirty   ❽
                                      MC error info valid

CAP Error Register        xE0000000   Uncorrectable ECC err det by MDPA ❸
                                      Uncorrectable ECC err det by MDPB

                                      MC error info latched             ❹

Sys Environmental Regs    x00000000
PCI Bus Trans Error Adr   x00000000
MDPA Status Register      xC0000000   MDPA Status Register Data Not Valid
MDPA Error Syndrome Reg   x00080089   MDPA Syndrome Register Data Not
Valid
MDPB Status Register      x80000000   MDPB Status Register Data Not Valid
MDPB Error Syndrome Reg   x000D00D1   MDPB Syndrome Register Data Not
Valid

PALcode Revision                      Palcode Rev: 1.21-3
```

5.3.4 MCHK 660 IOD-Detected Failure (System Bus Error)

The error log in Example 5-4 shows the following:

❶   CPU0 logged the error in a system with two CPUs.

❷   The External Interface Status Register does not record an error.

❸   Both IOD CAP Error Registers logged an error.

❹   The MC Error Info Registers 0 and 1 captured the error information.

❺   The commander at the time of the error was CPU3 (known from MC_ERR1).

❻   The command on the bus at the time was a write-back memory command.

Since this is an MCHK 660, the IOD detected the error on the bus, and CPU0 is logging the error. CPU0 registers are not important in this case since it is servicing the IOD interrupt. There are three devices that can put data on the system bus: CPUs, memory, or an IOD. From MC_ERR Register 1 we know that at the time of the error CPU3 put bad data on the bus while writing to memory. See Section 5.4 for a procedure designed to help with IOD-detected errors.

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Example 5-4 MCHK 660 IOD-Detected Failure (System Bus Error)

```
Logging OS                        2. DIGITAL UNIX
System Architecture               2. Alpha
Event sequence number             6.
Timestamp of occurrence              04-APR-1996 17:20:04
Host name                            whip16

System type register     x00000016  AlphaStation 4x00
Number of CPUs (mpnum)   x00000002

CPU logging event (mperr) x00000000                              ❶
Event validity                    1. O/S claims event is valid
Event severity                    1. Severe Priority
Entry type                      100. CPU Machine Check Errors

CPU Minor class                   2. 660 Entry

Software Flags           x0000000300000000
                                     IOD 1 Register Subpkt Pres
                                     IOD 2 Register Subpkt Pres
Active CPUs              x00000003
Hardware Rev            x00000000
System Serial Number             C1563
Module Serial Number
Module Type               x0000
System Revision         x00000000

* MCHK 660 Regs *
Flags:                  x00000000
PCI Mask                  x0000
Machine Check Reason      x0202
PAL SHADOW REG 0        x00000000
.
.
.
PAL SHADOW REG 7        x00000000
PALTEMP0               x0000000007
.
.
.
PALTEMP23              x00000000047FDA58
Exception Address Reg  xFFFFFC000038D784
                                     Native-mode instruction
                                     Exception PC  x3FFFFF00000E35E1
Exception Summary Reg  x00000000
Exception Mask Reg     x00000000
PAL BASE               x00000000020000
                                     Base addr for palcode =
x0000000008
Interrupt Summary Reg  x00000000200000
                                     EXT. HW interrupt at IPL21
                                     AST requests 3 - 0  x00000000
IBOX Ctrl and Status Reg  x000000C160000000
                                     Timeout Bit Not Set
                                     PAL Shadow Registers Enabled
                                     Correctable Err Intrpts Enabled
                                     ICACHE BIST Successful
                                     TEST_STATUS_H Pin Asserted
Icache Par Err Stat Reg  x00000000
Dcache Par Err Stat Reg  x00000000
Virtual Address Reg      xFFFFFFFFFFF800130
Memory Mgmt Flt Sts Reg  x00000000014990
                                     Ref resulted in DTB miss
```

```
                                   RA Field  x0000000006

                                   Opcode Field    x00000000000029
Scache Address Reg          xFFFFFF0000024EAF
Scache Status Reg           x00000000
Bcache Tag Address Reg      xFFFFFF80FFED6FFF
                                   Parity for ds and v bits
                                   Cache block dirty
                                   Cache block valid
                                   Tag address<38:20> is
x00000000000FFE
Ext Interface Address Reg xFFFFFF00FC00000F
Fill Syndrome Reg           x0000000000C5D2
Ext Interface Status Reg  xFFFFFFF004FFFFFF

                                   Error occurred during D-ref fill  ❷
LD LOCK                     xFFFFFF000020065F


** IOD SUBPACKET -> **             IOD 0 Register Subpacket
WHOAMI                      x000000BA  Device ID  x0000003A
                                   Bcache Size = 2MB
                                   VCTY ASIC Rev = 0
                                   Module Revision  0.
Base Address of Bridge      x000000F9E0000000
PCI Revision                x06008021  CAP Chip Revision  x00000001
                                   Horse Module Revision  x00000002
                                   Saddle Module Revision  x00000000
                                   Saddle Module Type  Left Hand
                                   EISA Present
                                   PCI Class Code  x00000600
MC-PCI Command Register     x06480FF1  Selftest passed
                                   Delayed read enabled
                                   Bridge PCI trans enabled
                                   Req 64 bit data trans enabled
                                   Accept 64 bit data trans enabled
                                   Check PCI Addr Parity enabled
                                   Check MC bus CMS/Addr Parity
                                   enabled
                                   Check MC bus NXM enabled
                                   Check all transaction enabled
                                   16 byte aligned block write enabled
                                   Write Pend Number Thresho x00000008
                                   RD_TYPE   Short
                                   RL_TYPE   Medium
                                   RM_TYPE   Long
                                   ARB_MODE  MC-PCI Bridge Priority
                                   Mode
Memory Host Addr Exten      x00000000
IO Host Addr Extension      x00000000
Interrupt Control           x00000003  MC-PCI Intr Enabled
                                   Device intr info enabled if en_int
                                   = 1
Interrupt Request           x00800000  Interrupts asserted  x00000000
                                   Hard Error
Interrupt Mask Register 0 x00C50010
Interrupt Mask Register 1 x00000000
MC Error Info Register 0  x4A26DBF0  MC bus trans addr <31:4>  x04A26DBF
MC Error Info Register 1  x800ED600  MC bus trans addr <39:32> x00000000

                                   MC_Command x00000016          ❻

                                   Device Id x0000003B           ❺
                                   MC error info valid

CAP Error Register          xA0000000  Uncorrectable ECC err det by MDPA  ❸

                                   MC error info latched         ❹
```

```
PCI Bus Trans Error Adr     x00000000
MDPA Status Register        x80000000  MDPA Chip Revision  x00000000
                                       MDPA Error Syndrome of
                                       uncorrectable read error
MDPA Error Syndrome Reg     x1E00001E  Cycle 0 ECC Syndrome
x0000000000001E

                                       Cycle 1 ECC Syndrome x00000000
                                       Cycle 2 ECC Syndrome x00000000
                                       Cycle 3 ECC Syndrome
x0000000000001E
MDPB Status Register        x00000000  MDPB Chip Revision  x00000000
MDPB Error Syndrome Reg     x00000000  Cycle 0 ECC Syndrome x00000000
                                       Cycle 1 ECC Syndrome x00000000
                                       Cycle 2 ECC Syndrome x00000000
                                       Cycle 3 ECC Syndrome x00000000

** IOD SUBPACKET -> **                 IOD 1 Register Subpacket
WHOAMI                      x000000BA  Device ID  x0000003A
                                       Bcache Size = 2MB
                                       VCTY ASIC Rev = 0
                                       Module Revision  0.
Base Address of Bridge      x000000FBE0000000
PCI Revision                x06000021  CAP Chip Revision  x00000001
                                       Horse ModuleRevision  x00000002
                                       Saddle Module Revision  x00000000
                                       Saddle Module Type  Left Hand
                                       PCI Class Code  x00000600
MC-PCI Command Register     x06480FF1  Selftest passed
                                       Delayed read enabled
                                       Bridge PCI trans enabled
                                       Req 64 bit data trans enabled
                                       Accept 64 bit data trans enabled
                                       Check PCI Addr Parity enabled
                                       Check MC bus CMS/Addr  Parity
                                       enabled
                                       Check MC bus NXM enabled
                                       Check all transaction enabled
                                       16 byte aligned block write enabled
                                       Write Pend Number Thresho x00000008
                                       RD_TYPE   Short
                                       RL_TYPE   Medium
                                       RM_TYPE   Long
                                       ARB_MODE  MC-PCI Bridge Priority
                                       Mode
Memory Host Addr Exten      x00000000
IO Host Addr Extension      x00000000
Interrupt Control           x00000003  MC-PCI Intr Enabled
                                       Device intr info enabled if en_int
                                       = 1
Interrupt Request           x00800000  Interrupts asserted  x00000000
                                       Hard Error
Interrupt Mask Register 0 x00C50001
Interrupt Mask Register 1 x00000000
MC Error Info Register 0  x4A26DBF0    MC bus trans addr <31:4>  x04A26DBF
MC Error Info Register 1  x800ED600    MC bus trans addr <39:32> x00000000

                                       MC_Command x00000016          ❻

                                       Device Id x0000003B           ❺
                                       MC error info valid

CAP Error Register        xA0000000     Uncorrectable ECC err det by MDPA  ❸

                                       MC error info latched          ❹
PCI Bus Trans Error Adr     x00000000
MDPA Status Register        x80000000  MDPA Chip Revision  x00000000
```

```
                                     MDPA Error Syndrome of
                                     uncorrectable read error
MDPA Error Syndrome Reg   x1E00001E  Cycle 0 ECC Syndrome x00000000
                                     Cycle 1 ECC Syndrome x00000000
                                     Cycle 2 ECC Syndrome x00000000
                                     Cycle 3 ECC Syndrome x00000000
MDPB Status Register      x00000000  MDPB Chip Revision  x00000000
MDPB Error Syndrome Reg   x00000000  Cycle 0 ECC Syndrome x00000000
                                     Cycle 1 ECC Syndrome x00000000
                                     Cycle 2 ECC Syndrome x00000000
                                     Cycle 3 ECC Syndrome x00000000
PALcode Revision                     Palcode Rev: 1.21-3
```

5.3.5 MCHK 660 IOD-Detected Failure (PCI Error)

The error log in Example 5-5 shows the following:

❶   CPU 0 logged the error in a system with three CPUs.

❷   The External Interface Status register records that the error occurred during a D-ref Fill but does not indicate what the error is.

❸   The CAP Error register for IOD0 did not see an error.

❹   The CAP Error register for IOD1, however, records a serious error.

❺   The MC Error Info registers 0 and 1 captured the error information.

❻   The commander at the time of the error was CPU0 and the command was a Read-IO (known from MC_ERR1).

❼   There is a PCI Subpacket from PCI1 with five nodes on it. Three devices on the PCI bus did not see an error, however two did, the Mylex DAC960 and the DEC_KZPSA. Either device could have caused the parity error.

Since this is an MCHK 660, the IOD-detected the error on the bus, and CPU0 is logging the error. CPU0 registers are not important in this case since it is servicing the IOD interrupt. There are three devices that can put data on the system bus: CPUs, memory, or an IOD. The CAP Error register for IOD1 saw a serious error and the MC Error Info register captured error information. See Section 5.4 for a procedure designed to help with IOD-detected errors.

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Esample 5-5   MCHK 660 IOD-Detected Failure (PCI Error)

```
Logging OS                        2. DIGITAL UNIX
System Architecture               2. Alpha
Event sequence number             2.
Timestamp of occurrence              27-AUG-1996 08:15:41
Host name                            mason3

System type register      x00000016  AlphaStation 4x00
Number of CPUs (mpnum)    x00000003
CPU logging event (mperr) x00000000       ❶

Event validity                    1. O/S claims event is valid
Event severity                    1. Severe Priority
Entry type                      100. CPU Machine Check Errors

CPU Minor class                   2. 660 Entry

Software Flags            x0000002300000000
                                     IOD 0 Register Subpkt Pres
                                     IOD 1 Register Subpkt Pres
                                     PCI 1 Bus Snapshot Present
Active CPUs               x00000007
Hardware Rev             x00000000
System Serial Number                 NI62503MWE
Module Serial Number
Module Type                  x0000
System Revision          x00000000

* MCHK 660 Regs *
Flags:                   x00000000
PCI Mask                     x0002
Machine Check Reason         x0202  IOD-Detected Hard Error -OR-
                                    DTag Parity Error (If Cached CPU)
PAL SHADOW REG 0         x0000000000000000
PAL SHADOW REG 1         x0000000000000000
PAL SHADOW REG 2         x0000000000000000
PAL SHADOW REG 3         x0000000000000000
PAL SHADOW REG 4         x0000000000000000
PAL SHADOW REG 5         x0000000000000000
PAL SHADOW REG 6         x0000000000000000
PAL SHADOW REG 7         x0000000000000000
PALTEMP0                 xFFFFFFFFB589C000
PALTEMP1                 x0000000000000000
PALTEMP2                 xFFFFFC000043CDA0
PALTEMP3                 x0000000000007C00
PALTEMP4                 x0000000000000003
PALTEMP5                 x0000000000000000
PALTEMP6                 x000000000001C6AF
PALTEMP7                 xFFFFFC000043C820
PALTEMP8                 x1F1E171515020100
PALTEMP9                 xFFFFFC000043CB10
PALTEMP10                xFFFFFC0000433E0C
PALTEMP11                xFFFFFC000043C970
PALTEMP12                xFFFFFC000043CD10
PALTEMP13                x0000000000026E80
PALTEMP14                x0000000000000000
PALTEMP15                x00000000000E0000
PALTEMP16                x0000020306600001
PALTEMP17                x0000000000000000
PALTEMP18                x0000000000000000
PALTEMP19                xFFFFFFFFB589F958
PALTEMP20                x00000000009D2000
```

```
PALTEMP21                   xFFFFFC000043CD40
PALTEMP22                   xFFFFFC000058D540
PALTEMP23                   x000000007FC67A58
Exception Address Reg       xFFFFFC0000433E0C
                                       Native-mode Instruction
                                       Exception PC  x3FFFFF000010CF83
Exception Summary Reg       x0000000000000000
Exception Mask Reg          x0000000000000000
PAL Base Address Reg        x0000000000020000
                                       Base Addr for PALcode:
x0000000000000008
Interrupt Summary Reg       x0000000000200000
                                       External HW Interrupt at IPL21
                                       AST Requests 3-0:
x0000000000000000
IBOX Ctrl and Status Reg    x000000C160000000
                                       Timeout Counter Bit Clear.
                                       IBOX Timeout Counter Enabled.
                                       Floating Point Instructions will Cause
                                       FEN Exceptions.
                                       PAL Shadow Registers Enabled.
                                       Correctable Error Interrupts Enabled.
                                       ICACHE BIST (Self Test) Was Successful.
                                       TEST_STATUS_H Pin Asserted
Icache Par Err Stat Reg     x0000000000000000
Dcache Par Err Stat Reg     x0000000000000000
Virtual Address Reg         xFFFFFFFFB6ED3D38
Memory Mgmt Flt Sts Reg     x0000000000016211
                                       If Error, Reference Which Caused Was Write
                                       If Err, Reference Resulted in DTB Miss
                                       Fault Inst RA Field:   x0000000000000008

                                       Fault Inst Opcode:   x000000000000002C
Scache Address Reg          xFFFFFF000002502F
Scache Status Reg           x0000000000000000
Bcache Tag Address Reg      xFFFFFF8077AFAFFF
                                       Last Bcache Access Resulted in a Miss.
                                       Value of Parity Bit for Tag Control Status
                                       Bits Dirty, Shared & Valid is Set.
                                       Value of Tag Control Dirty Bit is Clear.
                                       Value of Tag Control Shared Bit is Set.
                                       Value of Tag Control Valid Bit is Set.
                                       Value of Parity Bit Covering Tag Store
                                       Address Bits is Set.
Tag Address<38:20> Is:   x000000000000077A
Ext Interface Address Reg xFFFFFF007E00000F
Fill Syndrome Reg           x000000000000DE08

**Ext Interface Status Reg  xFFFFFFFF004FFFFFF**           ❷
                            **Error Occurred During D-ref Fill**
LD LOCK                     xFFFFFF0076750B8F

**** IOD SUBPACKET -> ****          **IOD 0 Register Subpacket**

WHOAMI                      x000008BA  Module Revision  2.
                                       VCTY ASIC Rev = 0
                                       Bcache Size = 2MB
                                       MID  2.
                                       GID  7.

Base Address of Bridge      x000000F9E0000000
Dev Type & Rev Register     x06008231  CAP Chip Revision:        x00000001
                                  HORSE  Module Revision:   x00000003
                                  SADDLE Module Revision:   x00000002
                                  SADDLE Module Type:         Left Hand
                                  PCI-EISA Bus Bridge Present on PCI Segment
```

```
                                PCI Class Code           x00000600
MC-PCI Command Register    x06470FB1  Module SelfTest Passed LED on
                                Delayed PCI Bus Reads Protocol: Enabled
                                Bridge to PCI Transactions: Enabled
                                Bridge WILL NOT REQUEST 64 Bit Data Trans
                                Bridge ACCEPTS 64 Bit Data Transactions
                                PCI Address Parity Check: Enabled
                                MC Bus CMD/Addr Parity Check: Enabled
                                MC Bus NXM Check: Enabled
                                Check ALL Transactions for Errors
                                Use MC_BMSK for 16 Byte Align Blk Mem Wrt
                                Wrt PEND_NUM Threshold:  7.
                                RD_TYPE Memory Prefetch Algorithm: Short
                                RL_TYPE Mem Rd Line Prefetch Type: Medium
                                RM_TYPE Mem Rd Multiple Cmd Type:  Long
                                ARB_MODE Arbitration: MC-PCI Priority Mode
Mem Host Address Ext Reg   x00000000  HAE Sparse Mem Adr<31:27> x00000000
IO Host Adr Ext Register   x00000000  PCI Upper Adr Bits<31:25> x00000000
Interrupt Ctrl Register    x00000003  Write Device Interrupt Info
Struct:Enabled
Interrupt Request          x00000000   Interrupts asserted  x00000000
Interrupt Mask0 Register   x00C50110
```

Interrupt Mask1 Register   x00000000   ❸

```
MC Error Info Register 0   xE0000000
                                MC Bus Trans Addr<31:4>: E0000000
MC Error Info Register 1   x000E89FD  MC bus trans addr <39:32> x000000FD
                                MC Command is Read1-IO
                                CPU0 Master at Time of Error
                                Device ID 2   x00000002
CAP Error Register         x00000000
Sys Environmental Regs     x00000000
PCI Bus Trans Error Adr    x00000000
MDPA Status Register       x00000000  MDPA Status Register Data Not Valid
MDPA Error Syndrome Reg    x00000000  MDPA Syndrome Register Data Not
Valid
MDPB Status Register       x00000000  MDPB Status Register Data Not Valid
MDPB Error Syndrome Reg    x00000000  MDPB Syndrome Register Data Not
Valid
```

**** IOD SUBPACKET -> ****   **IOD 1 Register Subpacket**

```
WHOAMI                     x000008BA  Module Revision  2.
                                VCTY ASIC Rev = 0
                                Bcache Size = 2MB
                                MID  2.
                                GID  7.


Base Address of Bridge     x000000FBE0000000
Dev Type & Rev Register    x06000231  CAP Chip Revision:       x00000001
                                HORSE  Module Revision:   x00000003
                                SADDLE Module Revision:   x00000002
                                SADDLE Module Type:        Left Hand
                                Internal CAP Chip Arbiter: Enabled
                                PCI Class Code           x00000600
MC-PCI Command Register    x06470FB1  Module SelfTest Passed LED on
                                Delayed PCI Bus Reads Protocol: Enabled
                                Bridge to PCI Transactions: Enabled
                                Bridge WILL NOT REQUEST 64 Bit Data Trans
                                Bridge ACCEPTS 64 Bit Data Transactions
                                PCI Address Parity Check: Enabled
                                MC Bus CMD/Addr Parity Check: Enabled
                                MC Bus NXM Check: Enabled
                                Check ALL Transactions for Errors
                                Use MC_BMSK for 16 Byte Align Blk Mem Wrt
                                Wrt PEND_NUM Threshold:  7.
```

```
                             RD_TYPE Memory Prefetch Algorithm: Short
                             RL_TYPE Mem Rd Line Prefetch Type: Medium
                             RM_TYPE Mem Rd Multiple Cmd Type:  Long
                             ARB_MODE Arbitration: MC-PCI Priority Mode
Mem Host Address Ext Reg  x00000000  HAE Sparse Mem Adr<31:27> x00000000
IO Host Adr Ext Register  x00000000  PCI Upper Adr Bits<31:25> x00000000
Interrupt Ctrl Register   x00000003  Write Device Interrupt Info
Struct:Enabled
Interrupt Request         x00800000  Interrupts asserted  x00000000
                                     Hard Error

Interrupt Mask0 Register  x00C51111
Interrupt Mask1 Register  x00000000
```

MC Error Info Register 0 xE0000000   ❺

                             MC Bus Trans Addr<31:4>: E0000000
MC Error Info Register 1 x000E89FD MC bus trans addr <39:32> x000000FD

                             MC Command is Read1-IO   ❻
                             CPU0 Master at Time of Error
                             Device ID 2 x00000002

CAP Error Register   x00000012 Serious error   ❹
                             PCI error address reg locked

```
Sys Environmental Regs    x00000000
PCI Bus Trans Error Adr   xC0BB6000
MDPA Status Register      x00000000  MDPA Status Register Data Not Valid
MDPA Error Syndrome Reg   x00000000  MDPA Syndrome Reg. Data Not Valid
MDPB Status Register      x00000000  MDPB Status Register Data Not Valid
MDPB Error Syndrome Reg   x00000000  MDPB Syndrome Reg. Data Not Valid


PALcode Revision                     Palcode Rev: 1.21-3
```

**** PCI SUBPACKET -> ****   **PCI 1 Subpacket**

```
Node Qty                         5.
```

CONFIG Address   x000000FBC0000800
Device and Vendor ID   x00011000 NCR 53C810 NCR_810 SCSI Narrow
SingleEnded

```
                                     Vendor ID:  x1000 (NCR)
                                     Device ID:  x00000001
Command Register       x0147  I/O Space Accesses Response:     Enabled
                       Memory Space Accesses Response:   Enabled
                       PCI Bus Master Capability:        Enabled
                       Monitor for Special Cycle Ops:    DISABLED
                       Generate Mem Wrt/Invalidate Cmds: DISABLED
                       Parity Error Detection Response:  Normal
                       Wait Cycle Address/Data Stepping: DISABLED
                       SERR# Sys Err Driver Capability:  Enabled
                       Fast Back-to-Back to Many Target: DISABLED
```
Status Register   x0200 Device is 33 Mhz Capable.
```
                       No Support for User Defineable Features.
                       Fast Back-to-Back to Different Targets,
                       Is Not Supported in Target Device.
                       Device Select Timing:  Medium.
Revision ID                   x02
Device Class Code             x010000  Mass Storage: SCSI Bus Controller
Cache Line S                  x00
Latency T.                    xFF
Header Type                   x00  Single Function Device
Bist                          x00
Base Address Register 1   x00101300
Base Address Register 2   x01119300
Base Address Register 3   x00000000
Base Address Register 4   x00000000
Base Address Register 5   x00000000
```

```
Base Address Register 6    x00000000
Expansion Rom Base Addres x00000000
Interrupt P1                        x04
Interrupt P2                        x01
Min Gnt                             x00
Max Lat                             x00



CONFIG Address             x000000FBC0001000
Device and Vendor ID       x00011069  Mylex DAC960 KZPSC RAID Controller
                                    Vendor ID:  x1069 (Mylex)
                                    Device ID:  x00000001
Command Register           x0147  I/O Space Accesses Response:     Enabled
                           Memory Space Accesses Response:   Enabled
                           PCI Bus Master Capability:        Enabled
                           Monitor for Special Cycle Ops:    DISABLED
                           Generate Mem Wrt/Invalidate Cmds: DISABLED
                           Parity Error Detection Response:  Normal
                           Wait Cycle Address/Data Stepping: DISABLED
                           SERR# Sys Err Driver Capability:  Enabled
                           Fast Back-to-Back to Many Target: DISABLED
Status Register                xC200  Device is 33 Mhz Capable.
                           No Support for User Defineable Features.
                           Fast Back-to-Back to Different Targets,
                           Is Not Supported in Target Device.
                           Device Select Timing:  Medium.
                           SIGNALED SYSTEM ERROR: This Device has Set
                           A System Error on SERR# Line.
                           DETECTED PARITY ERROR:This Device Detected
Revision ID                        x02
Device Class Code               x010400
Cache Line S                       x10
Latency T.                         xFF
Header Type                        x00   Single Function Device
Bist                               x00
Base Address Register 1    x00101200
Base Address Register 2    x01119200
Base Address Register 3    x00000000
Base Address Register 4    x00000000
Base Address Register 5    x00000000
Base Address Register 6    x00000000
Expansion Rom Base Addres x01110000
Interrupt P1                        x08
Interrupt P2                        x01
Min Gnt                             x04
Max Lat                             x00



CONFIG Address             x000000FBC0001800
Device and Vendor ID       x00011000  NCR 53C810 NCR_810 SCSI Narrow
SingleEnded
                                    Vendor ID:  x1000 (NCR)
                                    Device ID:  x00000001
Command Register           x0147  I/O Space Accesses Response:     Enabled
                           Memory Space Accesses Response:   Enabled
                           PCI Bus Master Capability:        Enabled
                           Monitor for Special Cycle Ops:    DISABLED
                           Generate Mem Wrt/Invalidate Cmds: DISABLED
                           Parity Error Detection Response:  Normal
                           Wait Cycle Address/Data Stepping: DISABLED
                           SERR# Sys Err Driver Capability:  Enabled
                           Fast Back-to-Back to Many Target: DISABLED
Status Register                x0200  Device is 33 Mhz Capable.
                           No Support for User Defineable Features.
```

```
                         Fast Back-to-Back to Different Targets,
                         Is Not Supported in Target Device.
                         Device Select Timing:  Medium.
Revision ID                      x02
Device Class Code             x010000  Mass Storage: SCSI Bus Controller
Cache Line S                     x00
Latency T.                       xFF
Header Type                      x00  Single Function Device
Bist                             x00
Base Address Register 1   x00101100
Base Address Register 2   x01119100
Base Address Register 3   x00000000
Base Address Register 4   x00000000
Base Address Register 5   x00000000
Base Address Register 6   x00000000
Expansion Rom Base Addres x00000000
Interrupt P1                     x0C
Interrupt P2                     x01
Min Gnt                          x00
Max Lat                          x00




CONFIG Address             x000000FBC0002000
Device and Vendor ID       x00091011  DECchip 21140 10/100Mhz TULIP
Ethernet
                              Vendor ID:  x1011 (Digital Equip Corp)
                              Device ID:  x00000009
Command Register     x0147  I/O Space Accesses Response:     Enabled
                         Memory Space Accesses Response:  Enabled
                         PCI Bus Master Capability:       Enabled
                         Monitor for Special Cycle Ops:   DISABLED
                         Generate Mem Wrt/Invalidate Cmds: DISABLED
                         Parity Error Detection Response:  Normal
                         Wait Cycle Address/Data Stepping: DISABLED
                         SERR# Sys Err Driver Capability:  Enabled
                         Fast Back-to-Back to Many Target: DISABLED
Status Register              x0280  Device is 33 Mhz Capable.
                         No Support for User Defineable Features.
                         Fast Back-to-Back to Different Targets,
                         Is Supported in Target Device.
                         Device Select Timing:  Medium.
Revision ID                      x12
Device Class Code     x020000  Network Controller: Ethernet Controller
Cache Line S                     x00
Latency T.                       xFF
Header Type                      x00  Single Function Device
Bist                             x00
Base Address Register 1   x00101000
Base Address Register 2   x01119000
Base Address Register 3   x00000000
Base Address Register 4   x00000000
Base Address Register 5   x00000000
Base Address Register 6   x00000000
Expansion Rom Base Addres x00000000
Interrupt P1                     x10
Interrupt P2                     x01
Min Gnt                          x00
Max Lat                          x00




CONFIG Address             x000000FBC0002800
Device and Vendor ID       x00081011  DEC_KZPSA  Fast-Wide-Differential SCSI
                         Vendor ID:  x1011 (Digital Equip Corp)
                              Device ID:  x00000008
```

```
Command Register         x0147  I/O Space Accesses Response:      Enabled
                                Memory Space Accesses Response:   Enabled
                                PCI Bus Master Capability:        Enabled
                                Monitor for Special Cycle Ops:    DISABLED
                                Generate Mem Wrt/Invalidate Cmds: DISABLED
                                Parity Error Detection Response:  Normal
                                Wait Cycle Address/Data Stepping: DISABLED
                                SERR# Sys Err Driver Capability:  Enabled
                                Fast Back-to-Back to Many Target: DISABLED
Status Register                 xE2C0  Device is 33 Mhz Capable.
                                Device Supports User Defineable Features.
                                Fast Back-to-Back to Different Targets,
                                Is Supported in Target Device.
                                Device Select Timing:  Medium.

                                RECEIVED MASTER-ABORT:Master Sets When Its    ❼
                                Transaction Terminated by MasterAbort.
                                SIGNALED SYSTEM ERROR: This Device has Set
                                A System Error on SERR# Line.
                                DETECTED PARITY ERROR:This Device Detected
Revision ID                     x00
Device Class Code               x010000  Mass Storage: SCSI Bus Controller
Cache Line S                    x10
Latency T.                      xFF
Header Type                     x00  Single Function Device
Bist                            x80
Base Address Register 1   x01118000
Base Address Register 2   x00000000
Base Address Register 3   x00100000
Base Address Register 4   x01000000
Base Address Register 5   x00000000
Base Address Register 6   x00000000
Expansion Rom Base Addres x01100000
Interrupt P1                    x14
Interrupt P2                    x01
Min Gnt                         x08
Max Lat                         x7F
```

5.3.6 MCHK 630 Correctable CPU Error

The error log in Example 5-6 shows the following:

❶   CPU0 logged the error in a system with two CPUs.

❷   During a D-ref fill, the External Interface Status Register shows no error but states that the "data source is b-cache." (When a CPU chip does not find data it needs to perform a task in any of its caches, it requests data from off the chip to fill its D-cache. It performs a D-ref fill.)

❸   Both IOD CAP Error Registers logged no error.

❹   The FIL Syndrome Register has a valid ECC code for the lower half of the data.

Machine check 630s are detected by CPUs when they either take data off the system bus or when they access their own B-cache. In this case, the data did not come from the system bus, otherwise bit <30> would be set in the External Interface Status Register. CPU0 had a single-bit, ECC correctable error.

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Example 5-6 MCHK 630 Correctable CPU Error

```
Logging OS                        2. DIGITAL UNIX
System Architecture               2. Alpha
Event sequence number           415.
Timestamp of occurrence           09-MAY-1996 14:56:30
Host name                         whip16

System type register    x00000016  AlphaStation 4x00
Number of CPUs (mpnum)  x00000002
CPU logging event (mperr) x00000000                        ❶

Event validity                    1. O/S claims event is valid
Event severity                    3. High Priority
Entry type                      100. CPU Machine Check Errors

CPU Minor class                   3.  Bcache error (630 entry)

Software Flags          x00000000
Active CPUs             x00000003
Hardware Rev            x00000000
System Serial Number              C1563
Module Serial Number
Module Type               x0000
System Revision         x00000000

Machine Check Reason      x0086  Alpha Chip Detected ECC Err, From
B-Cache
EI STAT                 xFFFFFFF004FFFFFF

                                  DATA SOURCE IS BCACHE  ❷
                        D-ref fill
                        EV5 Chip Rev 4
EI ADDRESS              xFFFFFF00138D85EF

FIL SYNDROME            x00000000000800                    ❹
ISR                     x0000000100200000

WHOAMI                  x00000000  Module Revision  0.
                                   MID  0.
                                   GID  0.

Sys Environmental Regs  x00000000
Base Addr of Bridge     x00000000
Dev Type & Rev Register x00000000  CAP Chip Revision:       x00000000
                                   Horse Module Revision:   x00000000
                                   Saddle Module Revision:  x00000000
                                   Saddle Module Type:      LeftHand
                                   Internal CAP Chip Arbiter: Enabled
                                   PCI Class Code           x00000000
MC Error Info Register 0 x00000000
                                   MC Bus Trans Addr<31:4>: 0
MC Error Info Register 1 x00000000  MC bus trans addr <39:32> x00000000
                                   MC Command is Illegal
                                   Illegal
                                   Device ID 2  x00000000

CAP Error Register      x00000000                          ❸
MDPA Status Register     x00000000  MDPA Status Register Data Not Valid
MDPA Error Syndrome Reg  x00000000  MDPA Syndrome Register Data Not Valid
MDPB Status Register     x00000000  MDPB Status Register Data Not Valid
MDPB Error Syndrome Reg  x00000000  MDPB Syndrome Register Data Not Valid
PALcode Revision                   Palcode Rev: 1.21-3
```

5.3.7 MCHK 620 Correctable Error

The MCHK 620 error is a correctable error detected by the IOD.

The error log in Example 5-7 shows the following:

❶   CPU0 logged the error in a system with two CPUs.

❷   The External Interface Status Register is not valid.

❸   The MC Error Info Registers 0 and 1 captured the error information.

❹   The commander at the time of the error was CPU0.

❺   The command at the time of the error was a write-back memory command.

The IOD detected a recoverable error on the system bus. The MC command at the time of the error is a WriteThru-Mem Command (x00000006). The system bus commander at the time of the error is CPU0. Since this is a write, the defective FRU is CPU0.

NOTE: The error log example has been edited to decrease its size; registers of interest are in bold type. The "Horse" module referred to in the error log is the system bus to PCI bus bridge module, the B3040 module. The "Saddle" module is the PCI motherboard, the B3050 module. The "MC" bus is the system bus.

Refer to Table 5-9 for information on decoding commands, and refer to Table 5-10 for information on node IDs.

Example 5-7   MCHK 620 Correctable Error

```
Logging OS                         2. DIGITAL UNIX
System Architecture                2. Alpha
Event sequence number             32.
Timestamp of occurrence               28-JUN-1996 19:45:42
Host name                             sect06

System type register      x00000016  AlphaStation 4x00
Number of CPUs (mpnum)    x00000002

CPU logging event (mperr) x00000000                        ❶

Event validity                     1. O/S claims event is valid
Event severity                     5. Low Priority
Entry type                       100. CPU Machine Check Errors


CPU Minor class                    4. 620 System Correctable Error


Software Flags            x0000000000000000
Active CPUs               x00000003
Hardware Rev              x00000000
System Serial Number              C1563
Module Serial Number
Module Type               x0000
```

```
System Revision              x00000000

Machine Check Reason          x0204  IOD Detected Soft Error
Ext Interface Status Reg  x0000000000000000
                                     Not Valid for 620 System    ❷
                                     Correctable Errors
Ext Interface Address Reg x0000000000000000
                                     Not Valid for 620 System
                                     Correctable Errors
Fill Syndrome Reg            x0000000000000000
                                     Not Valid for 620 System
                                     Correctable Errors
Interrupt Summary Reg        x0000000000000000
                                     Not Valid for 620 System
                                     Correctable Errors
WHOAMI                       x00000000  Module Revision  0.
                                        MID  0.
                                        GID  0.
Sys Environmental Regs       x00000000
Base Addr of Bridge          x000000FBE0000000
Dev Type & Rev Register      x06000032  CAP Chip Revision:      x00000002
                                        HORSE  Module Revision:  x00000003
                                        SADDLE Module Revision:  x00000000
                                        SADDLE Module Type:       LeftHand
                                        Internal CAP Chip Arbiter: Enabled
                                        PCI Class Code           x00000600
MC Error Info Register 0  x122D5640
                                     MC Bus Trans Addr<31:4>: 122D5640
MC Error Info Register 1  x800E9600  MC bus trans addr <39:32> x00000000
                                     MC Command is WriteBack Mem    ❺
                                     CPU0 Master at Time of Error

                                     Device ID 2  x00000002      ❹
                                     MC error info valid
CAP Error Register           x89000000  Error Detected but Not Logged

                                     Correctable ECC err det by MDPA   ❸
                                     MC error info latched
MDPA Status Register         x00000000  MDPA Status Register Data Not Valid
MDPA Error Syndrome Reg   x00000000  MDPA Syndrome Register Data Not Valid
MDPB Status Register         x00000000  MDPB Status Register Data Not Valid
MDPB Error Syndrome Reg   x00000000  MDPB Syndrome Register Data Not Valid
PALcode Revision                      Palcode Rev: 0.0-1
```

5.4 Troubleshooting IOD-Detected Errors

Step 1

Read the CAP Error Registers on both PCI bridges (F9E0000880 and FBE0000880). If one or both of these registers shows an error, match the register contents with the data pattern and perform the action indicated.

Table 5-3   CAP Error Register Data Pattern

Data Pattern	Most Likely Cause	Action
110x x00x x000 0000 0000 0000 000x xxxx	RDSB - Uncorrectable ECC error detected on upper QW of MC bus (D127:64>)	Go to Step 2
101x x00x x000 0000 0000 0000 000x xxxx	RDSA - Uncorrectable ECC error detected on lower QW of MC bus (D63:0>)	Go to Step 2
111x x00x x000 0000 0000 0000 000x xxxx	RDS detected in both QWs	Go t to Step 2
1001 1000 x000 0000 0000 0000 000x xxxx	CRDB - Correctable ECC error detected on upper QW of MC bus (D127:64>)	Go to Step 2
1000 0000 x000 0000 0000 0000 000x xxxx	CRDA - Correctable ECC error detected on lower QW of MC bus (D63:0>)	Go to Step 2
1001 1000 x000 0000 0000 0000 000x xxxx	CRD detected in both QWs.	
100x x10x x000 0000 0000 0000 000x xxxx	NXM - Nonexistent MC bus address	Go to Step 3
100x x01x x000 0000 0000 0000 000x xxxx	MC_ADR_PERR - MC bus address parity error	Go to Step 4
100x x00x 1000 0000 0000 0000 000x xxxx	PIO_OVFL - PIO buffer overflow	Go to Step 5
0000 0000 0000 0000 0000 0000 0001 1xxx	PTE_INV - Page table entry is invalid	Go to Step 6
0000 0000 0000 0000 0000 0000 0001 x1xx	MAB - Master abort	Go to Step 7
0000 0000 0000 0000 0000 0000 0001 xx1x	SERR - PCI system error	Go to Step 8
0000 0000 0000 0000 0000 0000 0001 xxx1	PERR - PCI parity error	Go to Step 9

5.4.1 System Bus ECC Error

Step 2

Read the MC_ERR1 register and match the contents with the data pattern. Perform the action indicated.

Table 5-4   System Bus ECC Error Data Pattern

MC_ERR1 Data Pattern	Most Likely Cause	Action
for Memory Read		
1000 0000 0000 xxxx xxxx 10xx 0xxx xxxx	Bad nondirty data from memory (bad memory)	Go to Step 10
1000 0000 0000 xxxx xxxx 111x 0xxx xxxx	Bad nondirty data from memory (bad memory)	Go to Step 10
1000 0000 0001 xxxx xxxx 10xx 0xxx xxx	Bad dirty data from a CPU	Replace CPU(s)
1000 0000 0001 xxxx xxxx 111x 0xxx xxxx	Bad dirty data from a CPU	Replace CPU(s)
for Memory or I/O Write		
1000 0000 000x xxx0 10xx 011x xxxx xxxx	Bad data from MID = 2	Replace CPU0
1000 0000 000x xxx0 11xx 011x xxxx xxxx	Bad data from MID = 3	Replace CPU1
1000 0000 000x xxx1 00xx 011x xxxx xxxx	Bad data from MID = 4	Replace IOD0
1000 0000 000x xxx1 01xx 011x xxxx xxxx	Bad data from MID = 5	Replace IOD0
1000 0000 000x xxx1 10xx 011x xxxx xxxx	Bad data from MID = 6	Replace CPU2 or IOD1
1000 0000 000x xxx1 11xx 011x xxxx xxxx	Bad data from MID = 7	Replace CPU3 or IOD1
for Memory Fill Transactions		
1000 0000 000x xxx1 00xx 110x xxxx xxxx	Bad data from MID = 4	Replace IOD0
1000 0000 000x xxx1 01xx 110x xxxx xxxx	Bad data from MID = 5	Replace IOD0
1000 0000 000x xxx1 10xx 110x xxxx xxxx	Bad data from MID = 6	Replace IOD1
1000 0000 000x xxx1 11xx 110x xxxx xxxx	Bad data from MID = 7	Replace IOD1

NOTE: IOD0 = B3040-AA bridge module; IOD1 = B3040-AB bridge module.

5.4.2 System Bus Nonexistent Address Error

Step 3

Determine which node (if any) should have responded to the command/address identified in MC_ERR1. Perform the action indicated.

Table 5-5   System Bus Nonexistent Address Error Troubleshooting

MC_ERR1 Data Pattern	Most Likely Cause	Action
1000 0000 000x xxxx xxxx xxxx 0xxx xxxx	Software generated an MC ADDR > TOP_OF_MEM reg	Fix software
1000 0000 0000 xxxx xxxx xxxx 1xxx 100x	PCI0 bridge did not respond	Replace IOD0
1000 0000 0000 xxxx xxxx xxxx 1xxx 101x	PCI1 bridge did not respond	Replace IOD0
1000 0000 0000 xxxx xxxx xxxx 1xxx 110x	PCI2 bridge did not respond	Replace IOD1
1000 0000 0000 xxxx xxxx xxxx 1xxx 111x	PCI3 bridge did not respond	Replace IOD1

NOTE: IOD0 = B3040-AA bridge module; IOD1 = B3040-AB bridge module.

5.4.3 System Bus Address Parity Error

Step 4

Determine which node put the bad command/adress on the system bus identified in
MC_ERR1. Perform the action indicated.

Table 5-6   Address Parity Error Troubleshooting

MC_ERR1 Data Pattern	Most Likely Cause	Action
1000 0000 000x xxx0 10xx xxxx xxxx xxxx	Data sourced by MID = 2	Replace CPU0
1000 0000 000x xxx0 11xx xxxx xxxx xxxx	Data sourced by MID = 3	Replace CPU1
1000 0000 000x xxx1 00xx xxxx xxxx xxxx	Data sourced by MID = 4	Replace IOD0
1000 0000 000x xxx1 01xx xxxx xxxx xxxx	Data sourced by MID = 5	Replace IOD0
1000 0000 000x xxx1 10xx xxxx xxxx xxxx	Data sourced by MID = 6	Replace CPU2 or IOD1
1000 0000 000x xxx1 11xx xxxx xxxx xxxx	Data sourced by MID = 7	Replace CPU3 or IOD1

NOTE: IOD0 = B3040-AA bridge module; IOD1 = B3040-AB bridge module.

5.4.4 PIO Buffer Overflow Error (PIO_OVFL)

Step 5

Enter the value of the CAP_CTRL register bits<19:16> (Actual_PEND_NUM) in the following formula. Compare the results as indicated in Table 5-7 to determine the most likely cause of the error. When an IOD is implicated in the analysis of the error, replace the one that capturered the error in its CAP Error Register.

Expected_PEND_NUM = 12 - ((2 * (X - 1)) + Y)

Where: X = Number of PCIs

Y = Number of CPUs

Table 5-7   Cause of PIO_OVFL Error

Comparison	Most Likely Cause	Action
Actual_PEND_NUM = Expected_PEND_NUM	Broken hardware on IOD	Replace IOD
Actual_PEND_NUM < Expected_PEND_NUM	Broken hardware on IOD	Replace IOD
Actual_PEND_NUM > Expected_PEND_NUM	PEND_NUM setup incorrect	Fix the software

NOTE: IOD0 = B3040-AA bridge module; IOD1 = B3040-AB bridge module.

5.4.5 Page Table Entry Invalid Error

Step 6

This error is almost always a software problem. However, if the software is known to be good and the hardware is suspected, swap the IOD.

5.4.6 PCI Master Abort

Step 7

Master aborts normally occur when the operating system is sizing the PCI bus. However, if the master abort occurs after the system is booted, read PCI_ERR1 and determine which PCI device should have responded to this PCI address. Replace this device.

5.4.7 PCI System Error

Step 8

For this error to occur a PCI device asserted SERR. Read the error registers in all the PCI devices to determine which device. The PCI device that set SERR should have information logged in its error registers that should indicate a device.

5.4.8 PCI Parity Error

Step 9

Read PCI_ERR1 and determine which PCI device normally uses that PCI address space. Replace that device. Also, read the error registers in all the PCI devices to determine which device was driving the PCI bus when the parity error occurred.

5.4.9 Broken Memory

Step 10

Refer to the following sections.

For a Read Data Substitute Error (uncorrectable ECC error)

When a read data substitute (RDS) error occurs, determine which memory module pair caused the error as follows:

1. Run the memory diagnostic to see if it catches the bad memory. If so, replace the memory module that it reports as bad.

2. At the SRM console prompt, enter the **show mem** command.

    ```
    P00>>> show mem
    ```

    This command displays the base address and size of the memory module pair for each slot.

    <p style="text-align:center">OR</p>

    Read the configuration packet, found in the error log, to retrieve the base address and size of the memory module pair.

3. Compare this address to the failing address from the MC_ERR1 and MC_ERR0 Registers to determine which memory slot is failing.

4. Replace both memory modules (high and low) for that slot. For an RDS error, there is no way to know which memory module (high or low) is bad.

For a Corrected Read Data Error (CRD)

When a CRD error occurs, determine which memory module pair caused the error as follows:

1. At the SRM console prompt, enter the **show mem** command. This command displays the base address and size of the memory module pair for each slot.

    ```
    P00>>> show mem
    ```

2. Compare this address to the failing address from the MC_ERR1 and MC_ERR0 Registers to determine which memory slot is failing.

3. When you have isolated the failing memory pair, determine which of the two modules is bad. (You cannot do this if the operating system is Windows NT.) Read the CPU FIL SYNDROME Register. If this register is non-zero, use the ECC syndrome bits in Table 5-8 to determine which module had the single-bit error.

Table 5-8 ECC Syndrome Bits Table

MDP Syndrome Values for Low-Order Memory								
01	02	04	08	10	20	40	80	CE
CB	D3	D5	D6	D9	DA	DC	23	25
26	29	2C	31	13	19	4F	4A	52
54	57	58	5B	5D	A2	A4	A8	B0
MDP Syndrome Values for High-Order Memory								
2A	34	0E	0B	15	16	1A	1C	E3
E5	E6	E9	EA	EC	F1	F4	A7	AB
AD	B5	8F	8A	92	94	97	98	9B
9D	62	64	67	68	6B	6D	70	75

5.4.10 Command Codes

Table 5-9 shows the codes for transactions on the system bus and how they are affected by the commander in charge of the bus during the transaction. The command is a six-bit field in the command address (bits<5:0>). Bit-to-text translations give six-bit data (although the top two bits may or may not be relevant). Note that address bit<39> defines the command as being either a system space or an I/O command.

Table 5-9 Decoding Commands

MC_CMD 5 4	MC_CMD 3 2 1 0	CMD in Hex	MC_ADR <39>	Description	No B-Cache CPU	B-Cache CPU	IOD
x x	0 0 0 0	X 0	1	Mem Idle		Y	Y
0 0	0 0 1 0	0 2	1	Write Pend Ack			Y
x x	0 0 1 1	X 3	1	Mem Refresh			
x x	0 1 0 1	X 4	0	Set Dirty		Y	
x 0	0 1 1 0	0/2 6	0	Write Thru - Mem	Y	Y	
x 0	0 1 1 0	0/2 6	1	Write Thru - I/O	Y	Y	
x 1	0 1 1 0	3/1 6	0	Write Back - Mem	Y	Y	
x 1	0 1 1 0	3/1 6	1	Write Intr - I/O			Y
0 0	0 1 1 1	0 7	0	Write Full - Mem			Y
1 0	0 1 1 1	2 7	0	Write Part - Mem (B-cache CPU only)			Y
x 0	0 1 1 1	0/2 7	1	Write Mask - I/O			Y
x 0	0 1 1 1	0/2 7	0	Write Merge - Mem			Y
x x	1 0 0 0	X 8	0	Read0 - Mem	Y	Y	Y
x x	1 0 0 0	X 8	1	Read0 - I/O	Y	Y	
x x	1 0 0 1	X 9	0	Read1 - Mem	Y	Y	Y
x x	1 0 0 1	X 9	1	Read1 - I/O	Y	Y	
x x	1 0 1 0	X A	0	Read Mod0 - Mem	Y	Y	Y

Table 5-9 Decoding Commands (continued)

MC_CMD 5 4	MC_CMD 3 2 1 0	CMD in Hex	MC_ADR <39>	Description	No B-Cache CPU	B-Cache CPU	IOD
x x	1 0 1 0	X A	0	Read Mod0 - Mem	Y	Y	Y
x x	1 0 1 0	X A	1	Read Peer0 - I/O			Y
x x	1 0 1 1	X B	0	Read Mod1 - Mem	Y	Y	Y
x x	1 0 1 1	X B	1	Read Peer1 - I/O			Y
1 0	1 1 0 0	2 C	1	FILL0 (due to Read0/Peer0)			Y
1 0	1 1 0 1	2 D	1	FILL1 (due to Read1/Peer1)			Y
x x	1 1 1 0	X E	0	Read0 - Mem	Y	Y	
x x	1 1 1 1	X F	0	Read1 - Mem	Y	Y	

5.4.11 Node IDs

The node ID is a six-bit field in the command address (bits<38:33>). The high-order three bits are always set, and the last three indicate the node. Bit-to-text translations give six-bit data, although only the last three bits define the node.

Table 5-10   Node IDs

Node ID <2:0>	Six Bit (Hex)	Node (4000)	Node (4100)
0 0 0	38		
0 0 1	39	Memory	Memory
0 1 0	3A	CPU0	CPU0
0 1 1	3B	CPU1	CPU1
1 0 0	3C	IOD0	IOD0
1 0 1	3D	IOD1	IOD1
1 1 0	3E	IOD2	CPU2
1 1 1	3F	IOD3	CPU3

5.5 Double Error Halts and Machine Checks While in PAL Mode

Two error cases require special attention. Neither double error halts or machine checks while the machine is in PAL mode result in error log entries. Nevertheless, information is available that can help determine what error occurred.

5.5.1 PALcode Overview

PALcode, privileged architecture library code, is used to implement a number of functions at the machine level without the use of microcode. This allows operating systems to make common calls to PALcode routines without knowing the hardware specifics of each system the operating system is running on. PALcode routines handle:

- Instructions that require complex sequencing, such as atomic operations
- Instructions that require VAX-style interlocked memory access
- Privileged instructions
- Memory management
- Context swapping
- Interrupt and exception dispatching
- Power-up initialization and booting
- Console functions
- Emulation of instructions with no hardware support

5.5.2 Double Error Halt

A double error halt occurs under the following conditions:

- A machine check occurs.

- PAL completes its tasks and returns control of the system to the operating system.

- A second machine check occurs before the operating system completes its tasks.

The machine returns to the console and displays the following message:

```
halt code = 6
double error halt
PC  =  20000004
Your system has halted due to an irrecoverable
error.  Record the error  halt code and PC and
contact your Digital Services representative.  In
addition, type INFO 5 and INFO 8 at the console and
record the results.
```

The **info 5** command (Example 5-9) causes the SRM console to read the PAL-built logout area that contains all the data used by the operating system to create the error entry.

The **info 8** command (Example 5-10) causes the SRM console to read the IOD 0 and IOD 1 registers.

5.5.3 Machine Checks While in PAL

If a machine check occurs while the system is running PALcode, PALcode returns to the SRM console, not to the operating system. The SRM console writes the following message:

```
halt code = 7
machine check while in PAL mode
PC  =  20000004
Your system has halted due to an irrecoverable
error.  Record the error halt code and PC and
contact your Digital Services representative.  In
addition, type INFO 3 and INFO 8 at the console and
record the results.
```

The **info 3** command (Example 5-8) causes the SRM console to read the "impure area," which contains the state of the CPU before it entered PAL.

Example 5-8   INFO 3 Command

```
P00>>> info 3
```

```
                          cpu00

per_cpu impure area       00004400
cns$flag                  00000001 : 0000
cns$flag+4                00000000 : 0004
cns$hlt                   00000000 : 0008
cns$hlt+4                 00000000 : 000c
cns$mchkflag              00000228 : 0210
cns$mchkflag+4            00000000 : 0214
cns$exc_addr              20000004 : 0318
cns$exc_addr+4            00000000 : 031c
cns$pal_base              00000000 : 0320
cns$pal_base+4            00000000 : 0324
cns$mm_stat               0000da10 : 0338
cns$mm_stat+4             00000000 : 033c
cns$va                    00080000 : 0340
cns$va+4                  00000002 : 0344
cns$icsr                  40000000 : 0348
cns$icsr+4                000000c1 : 034c
cns$ipl                   0000001f : 0350
cns$ipl+4                 00000000 : 0354
cns$ps                    00000000 : 0358
cns$ps+4                  00000000 : 035c
cns$itb_asn               00000000 : 0360
cns$itb_asn+4             00000000 : 0364
cns$aster                 00000000 : 0368
cns$aster+4               00000000 : 036c
cns$astrr                 00000000 : 0370
cns$astrr+4               00000000 : 0374
cns$isr                   00400000 : 0378
cns$isr+4                 00000000 : 037c
cns$ivptbr                00000000 : 0380
cns$ivptbr+4              00000002 : 0384
cns$mcsr                  00000000 : 0388
cns$mcsr+4                00000000 : 038c
cns$dc_mode               00000001 : 0390
cns$dc_mode+4             00000000 : 0394
cns$maf_mode              00000080 : 0398
cns$maf_mode+4            00000000 : 039c
cns$sirr                  00000000 : 03a0
cns$sirr+4                00000000 : 03a4
cns$fpcsr                 00000000 : 03a8
cns$fpcsr+4               ff900000 : 03ac
cns$icperr_stat           00000000 : 03b0
cns$icperr_stat+4         00000000 : 03b4
cns$pmctr                 00000000 : 03b8
cns$pmctr+4               00000000 : 03bc
cns$exc_sum               00000000 : 03c0
cns$exc_sum+4             00000000 : 03c4
cns$exc_mask              00000000 : 03c8
cns$exc_mask+4            00000000 : 03cc
cns$intid                 00000016 : 03d0
cns$intid+4               00000000 : 03d4
cns$dcperr_stat           00000000 : 03d8
cns$dcperr_stat+4         00000000 : 03dc
cns$sc_stat               00000000 : 03e0
cns$sc_stat+4             00000000 : 03e4
cns$sc_addr               000047cf : 03e8
cns$sc_addr+4             ffffff00 : 03ec
cns$sc_ctl                0000f000 : 03f0
cns$sc_ctl+4              00000000 : 03f4
cns$bc_tag_addr           ff7fefff : 03f8
cns$bc_tag_addr+4         ffffffff : 03fc
cns$ei_stat               04ffffff : 0400
cns$ei_stat+4             fffffff0 : 0404
```

```
cns$fill_syn              000000a7 : 0410
cns$fill_syn+4            00000000 : 0414
cns$ld_lock               0004eaef : 0418
cns$ld_lock+4             ffffff00 : 041c
```

Example 5-9   INFO 5 Command

```
P00>>> info 5
                                cpu00

per_cpu logout area             00004838
mchk$crd_flag                   00000320 : 0000
mchk$crd_flag+4                 00000000 : 0004
mchk$crd_offsets                00000118 : 0008
mchk$crd_offsets+4              00001328 : 000c
mchk$crd_mchk_code              00980000 : 0010
mchk$crd_mchk_code+4            00000000 : 0014
mchk$crd_ei_stat                eba00003 : 0018
mchk$crd_ei_stat+4              4143040a : 001c
mchk$crd_ei_addr                d1200067 : 0020
mchk$crd_ei_addr+4              47f90416 : 0024
mchk$crd_fill_syn               eba00003 : 0028
mchk$crd_fill_syn+4             d1200068 : 002c
mchk$crd_isr                    7ec38000 : 0030
mchk$crd_isr+4                  63ff4000 : 0034
mchk$flag                       00000320 : 0000
mchk$flag+4                     00000000 : 0004
```
mchk$isr   **00000000 : 0138**
mchk$isr+4   **00000000 : 013c**
mchk$icsr   **60000000 : 0140**
mchk$icsr+4   **000000c1 : 0144**
mchk$ic_perr_stat   **00000000 : 0148**
mchk$ic_perr_stat+4   **00000000 : 014c**
mchk$dc_perr_stat   **00000000 : 0150**
mchk$dc_perr_stat+4   **00000000 : 0154**
```
mchk$va                         ff8000a0 : 0158
mchk$va+4                       ffffffff : 015c
mchk$mm_stat                    000149d0 : 0160
mchk$mm_stat+4                  00000000 : 0164
mchk$sc_addr                    0001904f : 0168
mchk$sc_addr+4                  ffffff00 : 016c
mchk$sc_stat                    00000000 : 0170
mchk$sc_stat+4                  00000000 : 0174
mchk$bc_tag_addr                ff7fefff : 0178
mchk$bc_tag_addr+4              ffffffff : 017c
```
mchk$ei_addr   **066bc3ef : 0180**
mchk$ei_addr+4   **ffffff00 : 0184**
mchk$fill_syn   **000000a7 : 0188**

```
mchk$fill_syn+4                      00000000 : 018c
mchk$ei_stat                         04ffffff : 0190
mchk$ei_stat+4                       fffffff0 : 0194
mchk$ld_lock                         00005b6f : 0198
mchk$ld_lock+4                       ffffff00 : 019c


IOD: 0 base address: f9e0000000


WHOAMI:     0000003a    PCI_REV:    06008221
CAP_CTL:    02490fb1    HAE_MEM:    00000000    HAE_IO:     00000000
INT_CTL:    00000003    INT_REQ:    00800000    INT_MASK0:  00010000
INT_MASK1:  00000000    MC_ERR0:    e0000000    MC_ERR1:    800e88fd
CAP_ERR:    84000000    PCI_ERR:    00000000    MDPA_STAT:  00000000
MDPA_SYN:   00000000    MDPB_STAT:  00000000    MDPB_SYN:   00000000


IOD: 1 base address: fbe0000000


WHOAMI:     0000003a    PCI_REV:    06000221
CAP_CTL:    02490fb1    HAE_MEM:    00000000    HAE_IO:     00000000
INT_CTL:    00000003    INT_REQ:    00800000    INT_MASK0:  00010000
INT_MASK1:  00000000    MC_ERR0:    e0000000    MC_ERR1:    800e88fd
CAP_ERR:    84000000    PCI_ERR:    00000000    MDPA_STAT:  00000000
MDPA_SYN:   00000000    MDPB_STAT:  00000000    MDPB_SYN:   00000000
```

Example 5-10   INFO 8 Command

```
P00>>> info 8

IOD 0

WHOAMI:       0000003a   PCI_REV:      06008221
CAP_CTL:      02490fb1   HAE_MEM:      00000000   HAE_IO:       00000000
INT_CTL:      00000003   INT_REQ:      00000000   INT_MASK0:    00210000
INT_MASK1:    00000000   MC_ERR0:      e0000000   MC_ERR1:      000e88fd
CAP_ERR:      00000000   PCI_ERR:      00000000   MDPA_STAT:    00000000
MDPA_SYN:     00000000   MDPB_STAT:    00000000   MDPB_SYN:     00000000
INT_TARG:     0000003a   INT_ADR:      00006000   INT_ADR_EXT   00000000
                                                  :
PERF_MON:     00406ebf   PERF_CONT:    00000000   CAP_DIAG:     00000000
DIAG_CHKA:    10000000   DIAG_CHKB:    10000000   SCRATCH:      21011131
W0_BASE:      00100001   W0_MASK:      00000000   T0_BASE:      00001000
W1_BASE:      00800001   W1_MASK:      00700000   T1_BASE:      00008000
W2_BASE:      8000000    W2_MASK:      3ff00000   T2_BASE:      00000000
W3_BASE:      00000000   W3_MASK:      1ff00000   T3_BASE:      0000b800
W_DAC:        00000000   SG_TBIA:      00000000   HBASE:        00000000

IOD 1

WHOAMI:       0000003a   PCI_REV:      06000221
CAP_CTL:      02490fb1   HAE_MEM:      00000000   HAE_IO:       00000000
INT_CTL:      00000003   INT_REQ:      00000000   INT_MASK0:    00000000
INT_MASK1:    00000000   MC_ERR0:      e0000000   MC_ERR1:      000e88fd
CAP_ERR:      00000000   PCI_ERR:      00000000   MDPA_STAT:    00000000
MDPA_SYN:     00000000   MDPB_STAT:    00000000   MDPB_SYN:     00000000
INT_TARG:     0000003a   INT_ADR:      00006000   INT_ADR_EXT   00000000
                                                  :
PERF_MON:     004e31a6   PERF_CONT:    00000000   CAP_DIAG:     00000000
DIAG_CHKA:    10000000   DIAG_CHKB:    10000000   SCRATCH:      00000000
W0_BASE:      00100001   W0_MASK:      00000000   T0_BASE:      00001000
W1_BASE:      00800001   W1_MASK:      00700000   T1_BASE:      00008000
W2_BASE:      80000001   W2_MASK:      3ff00000   T2_BASE:      00000000
W3_BASE:      00000000   W3_MASK:      1ff00000   T3_BASE:      0000a000
W_DAC:        00000000   SG_TBIA:      00000000   HBASE:        00000000
```

Chapter 6

Error Registers

This chapter describes the registers used to hold error information. These registers include:

- External Interface Status Register
- External Interface Address Register
- MC Error Information Register 0
- MC Error Information Register 1
- CAP Error Register
- PCI Error Status Register 1

6.1 External Interface Status Register - EL_STAT

The EI_STAT register is a read-only register that is unlocked and cleared by any PALcode read. A read of this register also unlocks the EI_ADDR, BC_TAG_ADDR, and FILL_SYN registers subject to some restrictions. The EI_STAT register is not unlocked or cleared by reset.

Address   FF FFF0 0168

Type   R

```
 31 30 29 28 27      24 23    |        |        |        |        |        0
┌──┬──┬──┬──┬──────────┬───────────────────────────────────────────────────┐
│  │  │  │  │          │                      All 1s                        │
└──┴──┴──┴──┴──────────┴───────────────────────────────────────────────────┘
   │  │  │  │          └──────────────── CHIP_ID <3:0>
   │  │  │  └─────────────────────────── BC_TPERR
   │  │  └────────────────────────────── BC_TC_PERR
   │  └───────────────────────────────── EI_ES
   └──────────────────────────────────── COR_ECC_ERR
```

```
 61        |        |        |        |        |        36 35 34 33 32
┌──────────────────────────────────────────────────────┬──┬──┬──┬──┐
│                          All 1s                        │  │  │  │  │
└──────────────────────────────────────────────────────┴──┴──┴──┴──┘
                    SEO_HRD_ERR ──────────────────┘  │  │  │
                    FIL_IRD ──────────────────────────┘  │  │
                    EI_PAR_ERR ──────────────────────────┘  │
                    UNC_ECC_ERR ──────────────────────────────┘
```

PKW0453-96

Fill data from B-cache or main memory could have correctable or uncorrectable errors in ECC mode. In parity mode, fill data parity errors are treated as uncorrectable hard errors. System address/command parity errors are always treated as uncorrectable hard errors, irrespective of the mode. The sequence for reading, unlocking, and clearing EI_STAT, EI_ADDR, BC_TAG_ADDR, and FILL_SYN is as follows:

1. Read the EI_ADDR, BC_TAG_ADDR, and FIL_SYN registers in any order. Does not unlock or clear any register.

2. Read the EI_STAT register. This operation unlocks the EI_ADDR, BC_TAG_ADDR, and FILL_SYN registers. It also unlocks the EI_STAT register subject to conditions given in Table 6-2, which defines the loading and locking rules for external interface registers.

NOTE: If the first error is correctable, the registers are loaded but not locked. On the second correctable error, the registers are neither loaded nor locked.

Registers are locked on the first uncorrectable error except the second hard error bit. This bit is set only for an uncorrectable error that follows an uncorrectable error. A correctable error that follows an uncorrectable error is not logged as a second error. B-cache tag parity errors are uncorrectable in this context.

Table 6-1   External Interface Status Register

Name	Bits	Type	Description
COR_ECC_ERR	<31>	R	**Correctable ECC Error.** Indicates that fill data received from outside the CPU contained a correctable ECC error.
EI_ES	<30>	R	**External Interface Error Source.** When set, indicates that the error source is fill data from main memory or a system address/command parity error. When clear, the error source is fill data from the B-cache.
			This bit is only meaningful when <COR_ECC_ERR>, <UNC_ECC_ERR>, or <EI_PAR_ERR> is set in this register.
			This bit is not defined for a B-cache tag error (BC_TPERR) or a B-cache tag control parity error (BC_TC_ERR).
BC_TC_PERR	<29>	R	**B-Cache Tag Control Parity Error.** Indicates that a B-cache read transaction encountered bad parity in the tag control RAM.
BC_TPERR	<28>	R	**B-Cache Tag Address Parity Error.** Indicates that a B-cache read transaction encountered bad parity in the tag address RAM.
CHIP_ID	<27:24>	R	**Chip Identification.** Read as "4." Future update revisions to the chip will return new unique values.
	<23:0>		All ones.

Table 6-1 External Interface Status Register (continued)

Name	Bits	Type	Description
	<63:36>		All ones.
SEO_HRD_ERR	<35>	R	**Second External Interface Hard Error.** Indicates that a fill from B-cache or main memory, or a system address/command received by the CPU has a hard error while one of the hard error bits in the EI_STST register is already set.
FIL_IRD	<34>	R	**Fill I-Ref D-Ref.** When set, indicates that the error occurred during an I-ref fill. When clear, indicates that the error occurred during a D-ref fill. This bit has meaning only when one of the ECC or parity error bits is set.
			This bit is not defined for a B-cache tag parity error (BC_TPERR) or a B-cache tag control parity error (BC_TC_ERR).
EI_PAR_ERR	<33>	R	**External Interface Command/Address Parity Error.** Indicates that an address and command received by the CPU has a parity error.
UNC_ECC_ER	<32>	R	**Uncorrectable ECC Error**. Indicates that fill data received from outside the CPU contained an uncorrectable ECC error. In parity mode, this bit indicates a data parity error.

6.1.1 External Interface Address Register - EI_ADDR

The EI_ADDR register contains the physical address associated with errors reported by the EI_STAT register. It is unlocked by a read of the EI_STAT Register. This register is meaningful only when one of the error bits is set.

Address   FF FFF0 0148

Access   R

31							4	3	0
								All 1s	

61						40	39		32
		All 1s					EI_ADDR <39:32>		
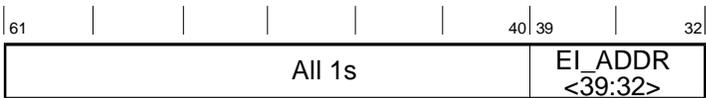
PKW0454-96

Table 6-2 Loading and Locking Rules for External Interface Registers

Correct-able Error	Uncorrect-able Error	Second Hard Error	Load Register	Lock Register	Action When EI_STAT Is Read
0	0	Not possible	No	No	Clears and unlocks all registers
1	0	Not possible	Yes	No	Clears and unlocks all registers
0	1	0	Yes	Yes	Clears and unlocks all registers
1[1]	1	0	Yes	Yes	Clear bit (c) does not unlock. Transition to "0,1,0" state.
0	1	1	No	Already locked	Clears and unlocks all registers
1[1]	1	1	No	Already locked	Clear bit (c) does not unlock. Transition to "0,1,1" state.

[1]These are special cases. It is possible that when EI_ADDR is read, only the correctable error bit is set and the registers are not locked. By the time EI_STAT is read, an uncorrectable error is detected and the registers are loaded again and locked. The value of EI_ADDR read earlier is no longer valid. Therefore, for the "1,1,x" case, when EI_STAT is read correctable, the error bit is cleared and the registers are not unlocked or cleared. Software must reexecute the IPR read sequence. On the second read operation, error bits are in "0,1,x" state, all the related IPRs are unlocked, and EI_STAT is cleared.

6.1.2 MC Error Information Register 0
(MC_ERR0 - Offset = 800)

The low-order MC bus (system bus) address bits are latched into this register when the system bus to PCI bus bridge detects an error event. If the event is a hard error, the register bits are locked. A write to clear symptom bits in the CAP Error Register unlocks this register. When the valid bit (MC_ERR_VALID) in the CAP Error Register is clear, the contents are undefined.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

| | 0 |

Failing Address ADDR<31:04>
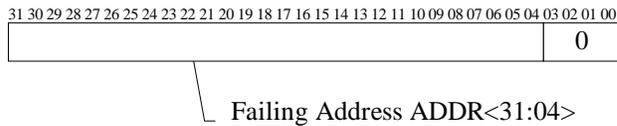
Table 6-3   MC Error Information Register 0

Name	Bits	Type	Initial State	Description
ADDR<31:4>	<31:4>	RO	0	Contains the address of the transaction on the system bus when an error is detected.
Reserved	<3:0>	RO	0	

6.1.3 MC Error Information Register 1 (MC_ERR1 - Offset = 840)

The high-order MC bus (system bus) address bits and error symptoms are latched into this register when the system bus to PCI bus bridge detects an error. If the event is a hard error, the register bits are locked. A write to clear symptom bits in the CAP Error Register unlocks this register. When the valid bit (MC_ERR_VALID) in the CAP Error Register is clear, the contents are undefined.

```
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
+--+--------------------------+-----+-----+--------------------------------+
|  |      reserved            | 1 1 | MID |                                |
+--+--------------------------+-----+-----+--------------------------------+
```

VALID

Dirty

DEVICE_ID

MC Command<5:0>

Failing Address ADDR<39:32>
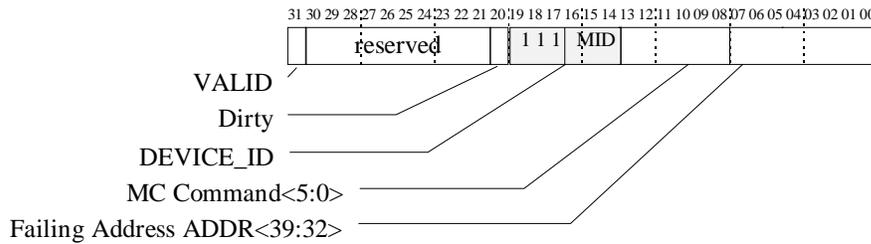
Table 6-4   MC Error Information Register 1

Name	Bits	Type	Initial State	Description
VALID	<31>	RO	0	Logical OR of bits <30:23> in the CAP_ERR Register. Set if MC_ERR0 and MC_ERR1 contain a valid address.
Reserved	<30:21>	RO	0	
Dirty	<20>	RO	0	Set if the system bus error was associated with a Read/Dirty transaction. When set, the device ID field <19:14> does not indicate the source of the data.
Reserved	<19:17>			All ones.
DEVICE_ID	<16:14>	RO	0	Slot number of bus master at the time of the error.
MC_CMD<5:0>	<13:8>	RO	0	Active command at the time the error was detected.
ADDR<39:32>	<7:0>	RO	0	Address bits <39:32> of the transaction on the system bus when an error is detected.

6.1.4 CAP Error Register
(CAP_ERR - Offset = 880)

CAP_ERR is used to log information pertaining to an error detected by the CAP or MDP ASIC. If the error is a hard error, the register is locked. All bits, except the LOST_MC_ERR bit, are locked on hard errors. CAP_ERR remains locked until the CAP error is written to clear each individual error bit.

Bit	Name	
	PERR	<0>
	SERR	<1>
	MAB	<2>
	PTE_INV	<3>
	PCI_ERR_VALID	<4>
	reserved	<22:5>
	PIO_OVFL	<23>
	LOST_MC_ERR	<24>
	MC_ADR_PERR	<25>
	NXM	<26>
	CRDA	<27>
	CRDB	<28>
	RDSA	<29>
	RDSB	<30>
	MC_ERR VALID	<31>
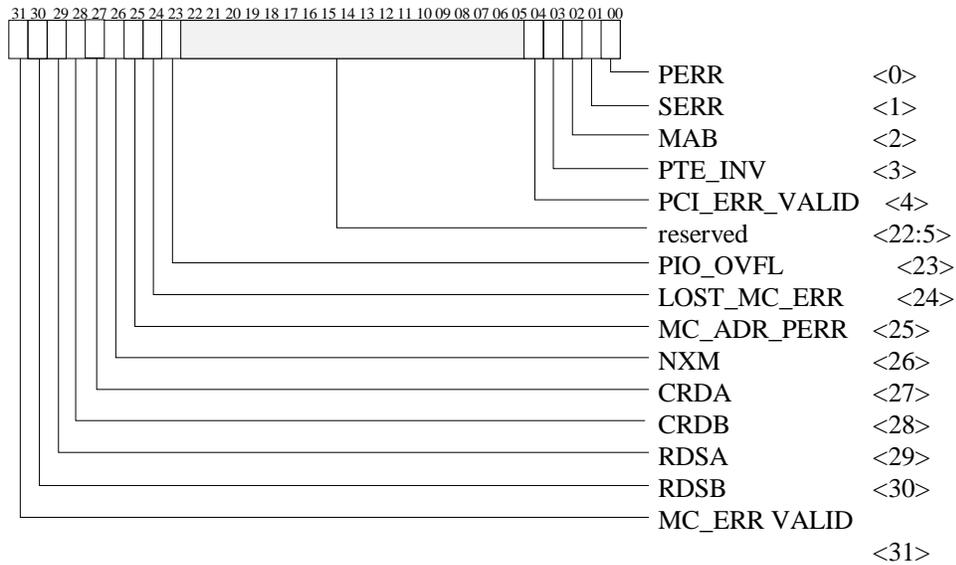
Table 6-5 CAP Error Register

Name	Bits	Type	Initial State	Description
MC_ERR VALID	<31>	RO	0	Logical OR of bits <30:23> in this register. When set MC_ERR0 and MC_ERR1 are latched.
RDSB	<30>	RW1C	0	Uncorrectable ECC error detected by MDPB. Clear state in MDPB before clearing this bit.
RDSA	<29>	RW1C	0	Uncorrectable ECC error detected by MDPA. Clear state in MDPA before clearing this bit.
CRDB	<28>	RW1C	0	Correctable ECC error detected by MDPB. Clear state in MDPB_STAT before clearing this bit.
CRDA	<27>	RW1C	0	Correctable ECC error detected by MDPA. Clear state in MDPA_STAT before clearing this bit.
NXM	<26>	RW1C	0	System bus master transaction status NXM (Read with Address bit <39> set but transaction not pended or transaction target above the top of memory register.) CPU will also get a fill error on reads.
MC_ADR_PERR	<25>	RW1C	0	Set when a system bus command/address parity error is detected.

Table 6-5 CAP Error Register (continued)

Name	Bits	Type	Initial State	Description
LOST_MC_ERR	<24>	RW1C	0	Set when an error is detected but not logged because the associated symptom fields and registers are locked with the state of an earlier error.
PIO_OVFL	<23>	RW1C	0	Set when a transaction that targets this system bus to PCI bus bridge is not serviced because the buffers are full. This is a symptom of setting the PEND_NUM field in CAP_CNTL to an incorrect value.
Reserved	<22:5>	RO	0	
PCI_ERR_VALID	<4>	RO	0	Logical OR of bits <3:0> of this register. When set, the PCI error address register is locked.
PTE_INV	<3>	RW1C	0	Invalid page table entry on scatter/gather access.
MAB	<2>	RW1C	0	PCI master state machine detected PCI Target Abort (likely cause: NXM) (except Special Cycle). On reads fill error is also returned.
SERR	<1>	RW1C	0	PCI target state machine observed SERR#. CAP asserts SERR when it is master and detects target abort.
PERR	<0>	RW1C	0	PCI master state machine observed PERR#.

6.1.5 PCI Error Status Register 1 (PCI_ERR1 - Offset = 1040)

PCI_ERR1 is used by the system bus to PCI bus bridge to log bus address <31:0> pertaining to an error condition logged in CAP_ERR. This register always captures PCI address <31:0>, even for a PCI DAC cycle. When the PCI_ERR_VALID bit in CAP_ERR is clear, the contents are undefined.

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

Failing Address ADDR<31:0>

Table 6-6 PCI Error Status Register 1

Name	Bits	Type	Initial State	Description
ADDR<31:0>	<31:0>	RO	0	Contains address bits <31:0> of the transaction on the PCI bus when an error is detected.

Chapter 7

Removal and Replacement

This chapter describes removal and replacement procedures for field-replaceable units (FRUs).

7.1 System Safety

Observe the safety guidelines in this section to prevent personal injury.

CAUTION: Wear an antistatic wrist strap whenever you work on a system. The AlphaServer cabinet system has a wrist strap connected to the frame at the front and rear. The pedestal system does not have an attached strap, so you will have to take one to the site.

WARNING: When the system interlocks are disabled and the system is still powered on, voltages are low in the system drawer, but current is high. Observe the following guidelines to prevent personal injury.

1. *Remove any jewelry that may conduct electricity before working on the system.*
2. *Do not insert your hands between the fan and the power supply.*
3. *If you need to access the system card cage, power down the system and wait 2 minutes to allow components in that area to cool.*

7.2 FRU List

Figure 7-1 shows the locations of FRUs in the system drawer, and Table 7-1 lists the part numbers of all field-replaceable units.

Figure 7-1 System Drawer FRU Locations

PKW0452-96
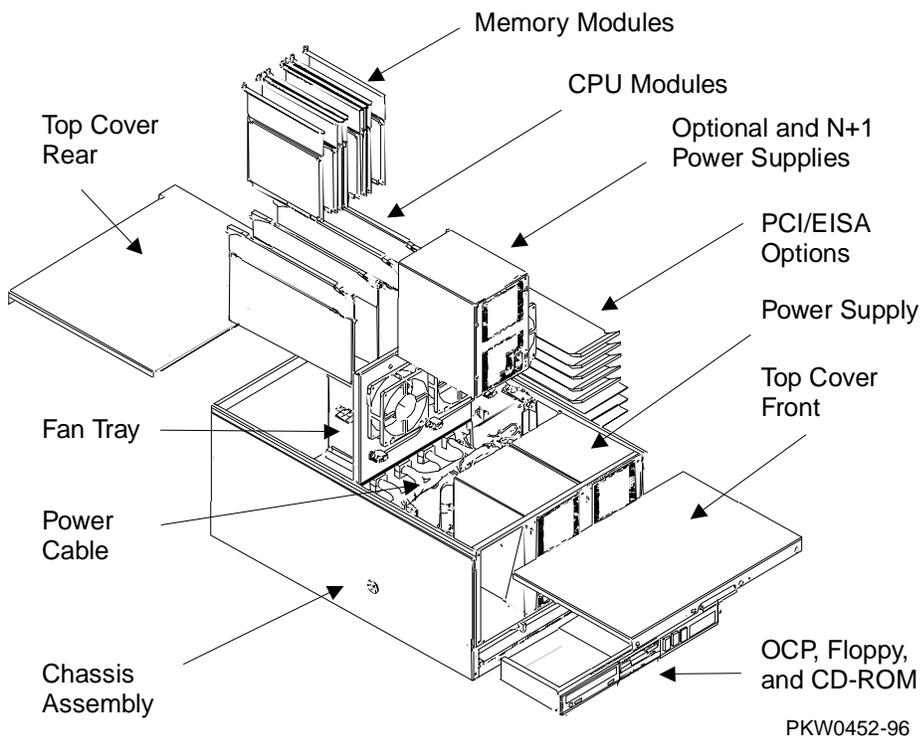
Table 7-1 Field-Replaceable Unit Part Numbers

CPU Modules

B3001-CA	300 MHz CPU, uncached
B3002-AB	300 MHz CPU, 2 Mbyte cache
B3004-BA	300 MHz CPU, 2 Mbyte cache
B3004-AA	400 MHz, 4 Mbyte cache
B3004-DA	466 MHz, 4 Mbyte cache

Memory Modules

B3020-CA	64 Mbyte synch
B3030-EA	256 Mbyte asynch (EDO)
B3030-FA	512 Mbyte asynch (EDO)
B3030-GA	2 Gbyte asynch (EDO)

Required System Drawer Modules and Display

54-23803-01	System motherboard (4100)
54-23803-02 54-23805-01	System motherboard (early 4000) System motherborard (4000)
B3040-AA	System bus to PCI bus bridge module (both systems)
B3040-AB	System bus to PCI bus bridge module (later 4000 only)
54-24117-01	Power control module
B3050-AA	PCI motherboard (both systems)
B3051-AA	PCI motherboard (later 4000 only)
54-24364-01	OCP logic module
54-24366-01	OCP switch module
54-24674-01	Server control module
54-24691-01	Fan fail detect module (cabinet only)
30-43049-01	OCP display

Table 7-1 Field-Replaceable Unit Part Numbers (continued)

Fans	
12-23609-21	4.5-inch fan
12-24701-34	CPU fan

Power System Components	
30-44712-01	Power supply (H7291-AA)
30-45353-01	Techniq AC Box (NA/Japan, H9A10-EB cabinet)
30-45353-02	Techniq AC Box (Europe/AP, H9A10-EC cabinet)
30-46788-01	Internal power source 40W/12V fan tray power (cabinet)
H7600-AA	Power controller (NA/Japan, H9A10-EL cabinet)
H7600-DB	Power controller (Europe/AP, H9A10-EM cabinet)
12-23501-01	NEMA power strip (N.A./Japan, pedestal)
12-45334-02	IEC power strip (Europe/AP, pedestal, and all cabinet systems)

Internal Power Cords	
17-04285-01	.5 meter IEC to IEC
17-00606-02	6 foot NEMA to IEC (N.A./Japan, pedestal)
17-04285-02	2 meter IEC to IEC (Europe/AP, pedestal, and all cabinet systems.)
17-04285-03	IEC to IEC StorageWorks shelf

Fan Tray Cables (Cabinet Only)	
17-04324-01	Elec fan power harness
17-04325-01	12V power for SCM
17-04338-01	Power ground cable
17-04339-01	AC cable power

Table 7-1 Field-Replaceable Unit Part Numbers (continued)

Server Control Module Power (Pedestal Only)	
30-46485-01	110V North America
30-46485-02	220V Europe
30-46485-03	Australia/N.Z.
30-46485-04	220V U.K.

System Drawer Cables and Jumpers		From	To
17-04196-01	Server control module signal cable (60 pin)	Remote I/O signal conn on PCI mbrd	SCM signal conn
17-04199-01	Current share cable	Current share conn on PS0	Current share conn on PS1 and PS2
17-04200-01	Floppy signal cable (36 pin)	Floppy conn on PCI mbrd	Floppy
17-04201-01	OCP signal	OCP conn on PCI mbrd	OCP signal (system drawer only)
17-04201-02	OCP signal jumper	OCP	OCP
17-04217-01	Power harness (4100 & early 4000)	Power supply(s)	7 conns. sys mbrd sys fans 0, 1 5V conn on PCI mbrd CD-ROM drv pwr Floppy pwr 1 OCP DC enable pwr conn or pwr conn on ped tray pwr drive cable (17-04293-01)

Table 7-1 Field-Replaceable Unit Part Numbers (continued)

System Drawer Cables and Jumpers		From	To
17-04358-01	Power harness (later 4000 only)	Power supply(s)	3 conns. sys mbrd sys fans 0, 1 5V conn on PCI/EISA mbrd 5V & 3V conn on PCI (right) mbrd CD-ROM drv pwr Floppy pwr 1 OCP DC enable pwr conn or pwr conn on ped tray pwr drive cable (17-04293-01)
17-04292-01	SCSI CD-ROM sig cable	CD-ROM conn on PCI mbrd	CD-ROM sig conn
70-32016-01	Interlock switches and cable (4100 & early 4000)	Interlock switch assy	Other OCP DC enable pwr conn or pwr conn on ped tray pwr drive cable (17-04293-01)
70-33002-01	Interlock switches and cable (later 4000 only)	Interlock switch assy	Other OCP DC enable pwr conn or pwr conn on ped tray pwr drive cable (17-04293-01)
17-04349-01	SCM 12V interlock jumper	Interlock conn on PCI mbrd	12 V DC enable conn on SCM
17-04350-01	SCM 34-position jumper	SCM	SCM
17-04351-01	SCM 12V power jumper (4100 & early 4000 only)	Power harness (17-04217-01)	Sys fan 2 and SCM internal 12V conn
17-04363-01	SCM 16-position jumper	SCM sig conn on PCI mbrd	16 pos conn on SCM

Table 7-1 Field-Replaceable Unit Part Numbers (continued)

Pedestal Cables		From	To
17-04293-01	Elec harness power cable+5/+12	Power harness (17-04217-01)	Ped tray bulkhead (system side)
17-04302-01	OCP signal cable	OCP sig conn on PCI mbrd	OCP sig conn on ped tray bulkhead (system side)
17-04305-01	Harness power cable +5/+12	Power conn on ped tray bulkhd (tray side)	Both OCP DC enable pwr conn and pwr conn on optional SCSI drive
17-04306-01	SCSI signal cable (narrow)	SCSI sig conn on ped tray bulkhd (tray side)	Optional SCSI drive
17-04380-01	OCP signal cable	OCP sig conn on ped tray bulkhd	OCP sig conn

7.3 4100 Power System FRUs

Figure 7-2   Location of 4100 Power System FRUs
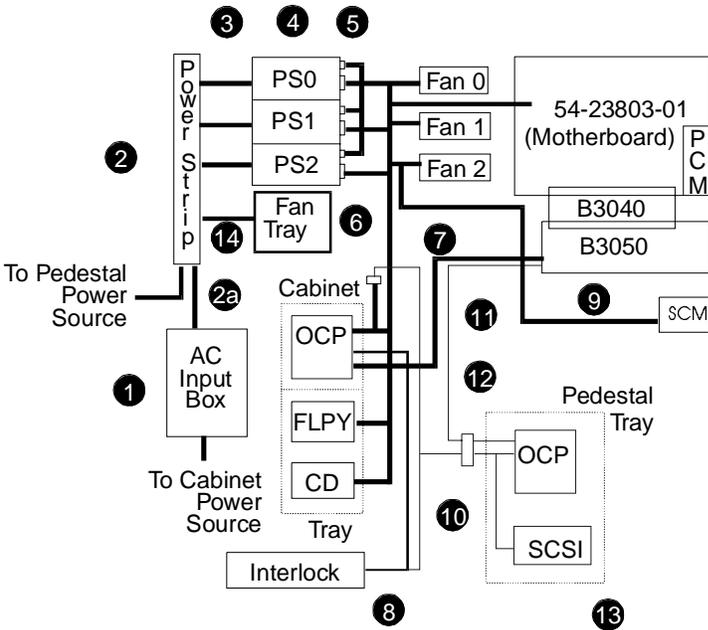
Notes: Only power cables are shown. Systems
have only one OCP located in either the cabinet tray
or the pedestal tray. Thicker lines indicate cables
present in both cabinets. Thinner lines are cables in
the pedestal only.

PKW-0401a-96

	Part Number	Description
1	30-45353-01	AC input box; only in cabinet systems: The 01 variant is for N.
	30-45353-02	Amer./Japan and has a NEMA L6-30P power cord; the 02 variant is for Europe and AP and has an IEC 309 power cord.
2	12-23501-01	AC power strip: The 12-23501-01 is used on pedestals in N.
	12-45334-02	Amer./Japan only and has six NEMA outlets and a 15 ft. cord to the wall outlet; the 12-45334-02 is used on pedestals in Eur./AP and on cabinet systems worldwide and has six IEC320 outlets. In pedestal systems, cords match country-specific wall outlets.

	Part Number	Description
2a	17-04285-01	Power cord from AC input box to power strip. .5 meter, IEC320 to IEC320 connector used in cabinet systems only. In pedestal systems, cords match country-specific wall outlets.
1, 2, 2a	H7600-AA	Power controller used in place of 30-45353-01, 12-45334-02, and 17-04285-02 in the H9A10-EL cabinet in N. America/Japan
1, 2, 2a	H7600-DB	Power controller used in place of 30-45353-02, 12-45334-02, and 17-04285-02 in the H9A10-EM cabinet in Europe/AP
3	17-00606-02 17-04285-02	Power cord from power strip to power supply: The 17-00606-02 is a 2 m NEMA to IEC320 AC jumper used with the 12-23501-01 power strip in N. Amer./Japan pedestals. The 17-04285-02 is a 2 m IEC320 to IEC320 AC jumper used with the 12-45334-02 power strip used on pedestals in Eur./APA and on cabinet systems worldwide and has six IEC320 outlets. In pedestal systems, cords match country-specific wall outlets.
4	30-44712-01	Power supply; 92 to 264 VAC input; one to three in a system drawer.
5	17-04199-01	Cable connecting power supplies
6	17-04217-01	Power distribution harness (4100 and early 4000)
7	17-04201-01	Cable from OCP to PCI motherboard (cabinet system)
8	70-32016-01	Interlock switches and cable to OCP (4100 and early 4000)
9	17-04351-01	Power from power harness between harness and Fan 2 to SCM
10	17-04293-01	Cable from power harness to interconnect cable and pedestal tray connector (pedestal system)
11	17-04302-01	Cable from pedestal tray connector to PCI motherboard (pedestal system)
12	17-04201-01	Cable from pedestal tray connector to OCP (pedestal system)
13	17-04305-01	Cable from pedestal tray connector to OCP and SCSI devices (pedestal system)
14	17-04339-01	Power cord from power strip to cabinet fan tray (cabinet only)

7.4 4000 Power System FRUs

Figure 7-3   Location of 4000 Power System FRUs
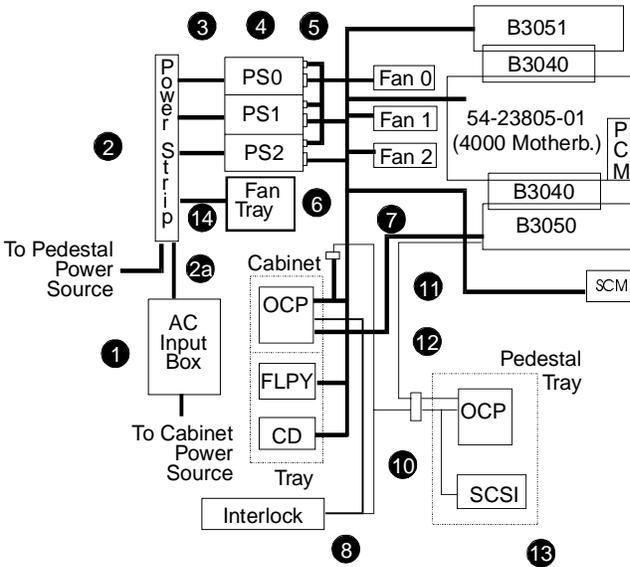
Note: Only power cables are shown. Systems have only one OCP located in either the cabinet tray or the pedestal tray. Thicker lines indicate cables present in both cabinets. Thinner lines are cables in the pedestal only. Callout 9 is intentionally missing.

PKW-0401C-96

	Part Number	Description
1	30-45353-01 30-45353-02	AC input box; only in cabinet systems: The 01 variant is for N. Amer./Japan and has a NEMA L6-30P power cord; the 02 variant is for Europe and AP and has an IEC 309 power cord.
2	12-23501-01 12-45334-02	AC power strip: The 12-23501-01 is used on pedestals in N. Amer./Japan only and has six NEMA outlets and a 15 ft. cord to the wall outlet; the 12-45334-02 is used on pedestals in Eur./AP and on cabinet systems worldwide and has six IEC320 outlets. In pedestal systems, cords match country-specific wall outlets.

	Part Number	Description
2a	17-04285-01	Power cord from AC input box to power strip. .5 meter, IEC320 to IEC320 connector used in cabinet systems only. In pedestal systems, cords match country-specific wall outlets.
1, 2, 2a	H7600-AA	Power controller used in place of 30-45353-01, 12-45334-02, and 17-04285-02 in the H9A10-EL cabinet in N. America/Japan
1, 2, 2a	H7600-DB	Power controller used in place of 30-45353-02, 12-45334-02, and 17-04285-02 in the H9A10-EM cabinet in Europe/AP
3	17-00606-02 17-04285-02	Power cord from power strip to power supply: The 17-00606-02 is a 2 m NEMA to IEC320 AC jumper used with the 12-23501-01 power strip in N. Amer./Japan pedestals. The 17-04285-02 is a 2 m IEC320 to IEC320 AC jumper used with the 12-45334-02 power strip used on pedestals in Eur./APA and on cabinet systems worldwide and has six IEC320 outlets. In pedestal systems, cords match country-specific wall outlets.
4	30-44712-01	Power supply; 92 to 264 VAC input; one to three in a system drawer.
5	17-04199-01	Cable connecting power supplies
6	17-04385-01	Power distribution harness (later 4000 only)
7	17-04201-01	Cable from OCP to PCI motherboard (cabinet system)
8	70-33002-01	Interlock switches and cable to OCP (later 4000 only)
10	17-04293-01	Cable from power harness to interconnect cable and pedestal tray connector (pedestal system)
11	17-04302-01	Cable from pedestal tray connector to PCI motherboard (pedestal system)
12	17-04201-01	Cable from pedestal tray connector to OCP (pedestal system)
13	17-04305-01	Cable from pedestal tray connector to OCP and SCSI devices (pedestal system)
14	17-04339-01	Power cord from power strip to cabinet fan tray (cabinet only)

7.5 System Drawer Exposure (Cabinet)

There are two cabinet types for these systems: the H9A10-EB -EC cabinet and the H9A10-EL -EM cabinet. System drawer exposure differs depending upon the cabinet.

7.5.1 Cabinet Drawer Exposure (H9A10-EB & -EC)

Open both doors, disconnect cables that obstruct movement of the drawer, remove the shipping brackets, and slide the drawer out from the cabinet.

Figure 7-4   Exposing System Drawer (H9A10-EB & -EC Cabinet)

Shipping Brackets
on Cabinet Rails

System Bus Cover

Power Section
Cover

PCI Bus
Cover

PKW0404-96
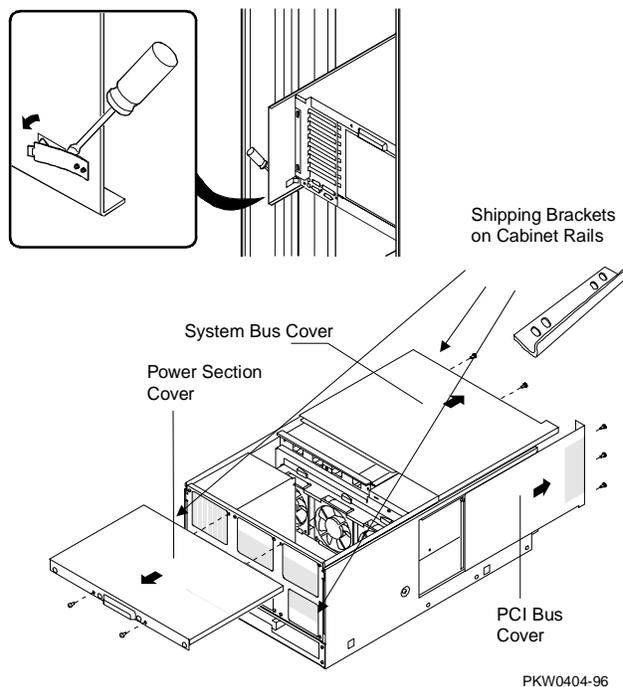
Exposing the System Bus or PCI Bus Card Cages

1. Open the front and rear doors of the cabinet.

2. At the front of the cabinet, unplug the drawer's power supplies.

3. At the rear, remove the two Phillips screws holding the shipping bracket on the right rail so that the drawer can be pulled out.

4. Using a flathead screwdriver, disengage the lock mechanism at the lower left hand corner of the drawer.

5. Pull the drawer out part way and release the lock mechanism by removing the screwdriver. If you wish to remove the whole drawer for some reason, leave the screwdriver in place.

6. Once the lock mechanism has been released, slide the drawer out until it locks.

7. Remove the system bus card cage cover. Unscrew the two Phillips head screws holding the cover in place and slide it off the drawer.

8. Remove the PCI bus card cage cover. Unscrew the three Phillips head screws holding the cover to the side of the drawer and slide it off the drawer.

Exposing the Power System or System Fans

1. Open the front and rear doors of the cabinet.

2. At the rear of the cabinet, remove any cables from PCI options that may interfere with pulling the drawer forward.

3. At the front, remove the shipping brackets on the right and left rails that hold the drawer.

4. Pull out the drawer until it locks.

5. Remove the power section cover. Unscrew the two Phillips head screws and slide the cover off the drawer.

7.5.2 Cabinet Drawer Exposure (H9A10-EL & EM)

In the H9A10-EL and -EM Cabinet, the system drawer sits on a tray that slides out of the front of the cabinet. A stabilizer bar must be pulled out from the bottom to pevent the cabinet from tipping over.

Figure 7-5   Exposing System Drawer (H9A10-EL & -EM Cabinet)
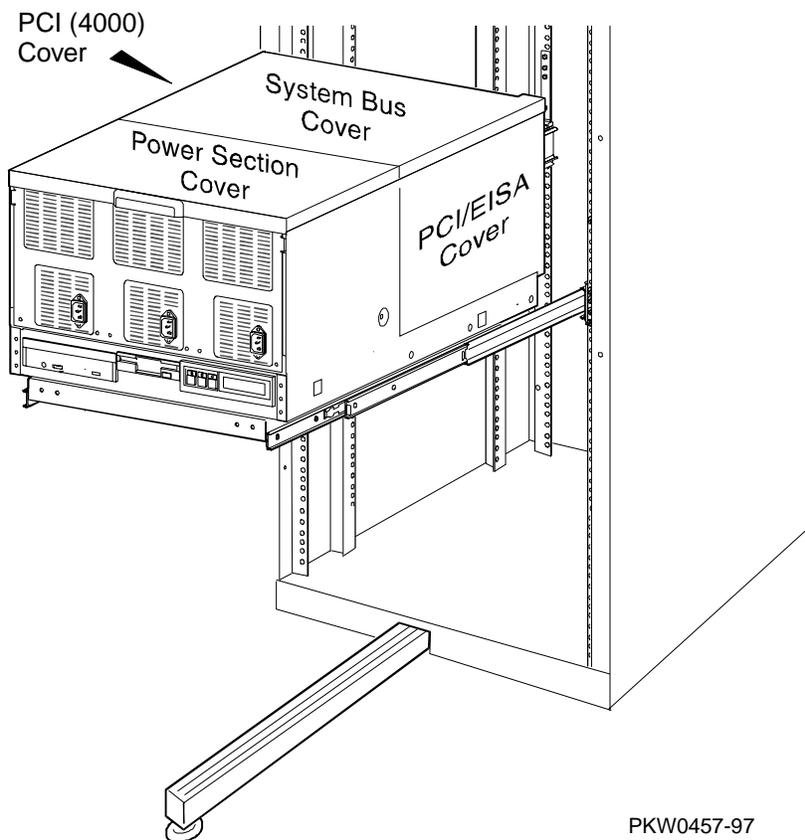
PKW0457-97

CAUTION: The cabinet could tip over if a system drawer is pulled out and the stablizing bar is not fully extended and its leveler foot on the floor.

Exposing any section of the system drawer in an H9A10-EL or -EM Cabinet.

1. Open the front door of the cabinet.

2. Pull the stabilizer bar at the bottom of the cabinet out until it stops.

3. Extend the leveler foot at the enc of the stabilizer bar to the floor.

4. Unplug the drawer's power supplies.

5. Remove the Phillips screws holding the shipping bracket to the rails so that the drawer can be pulled out.

6. Pull the drawer all the way out until it locks

7. To access the system bus card cage cover, unscrew the two Phillips head screws holding the cover in place and slide it off.

8. To access the PCI/EESA bus card cage, unscrew the three Phillips head screws holding the cover to the right side of the drawer and slide it off.

9. To access the PCI bus card cage, unscrew the three Phillips head screws holding the cover to the left side of the drawer and slide it off.

10. To access the power or fan section, unscrew the two Phillips head screws holding the cover in place and slide it off.

7.6 System Drawer Exposure (Pedestal)

Figure 7-5   Exposing System Drawer (Pedestal)

Pedestal Tray Cover

System Bus Cover

Pedestal Tray and Power Section Cover

PCI Bus Cover

PKW0429-96
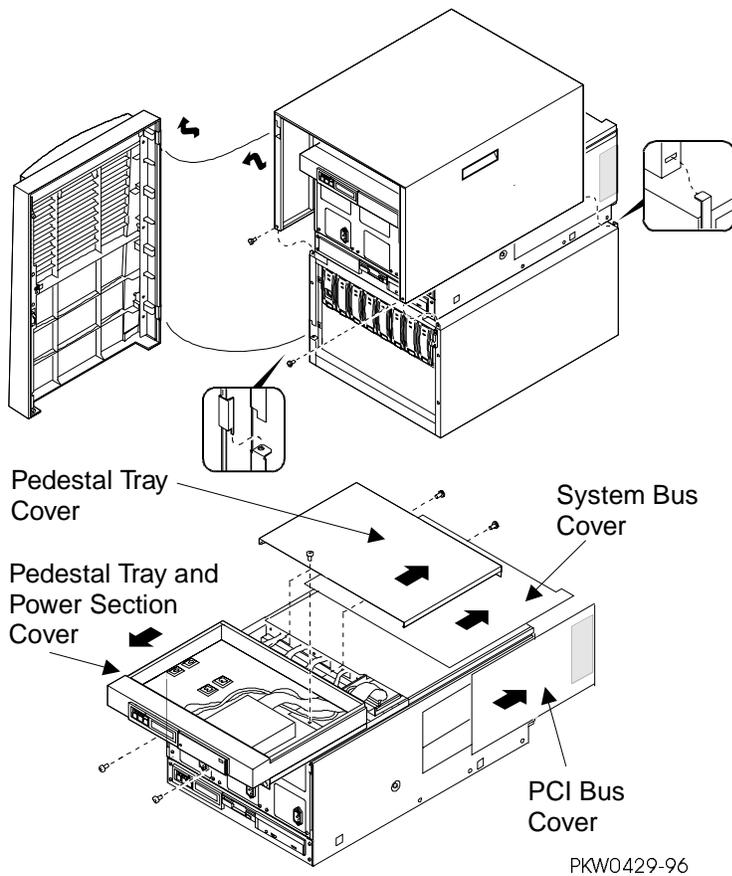
Exposing the System Drawer

1. Open the front door and remove it by lifting and pulling it away from the system.

2. Remove the top cover. Unscrew the two Phillips head screws midway up on each side of the pedestal, tilt the cover up, and lift it away from the frame.

3. Remove the system bus card cage cover at the back of the pedestal if you are replacing any of the following: CPU, memory, power control module, system bus to PCI bus module, system motherboard, cables that attach to the system motherboard, or a system fan. To remove the cover, unscrew the two Phillips head screws and slide the cover off the drawer.

4. Remove the PCI bus card cage cover at the back of the pedestal if you are replacing any of the following: PCI or EISA option, server control module, PCI motherboard, cables attached to the PCI motherboard. To remove the cover, unscrew the three Phillips head screws holding the cover to the side of the drawer and slide the cover off the drawer.

5. Remove the pedestal tray as described below if you are replacing any of the following: system fan, power supply, power cables.

Removing the Pedestal Tray

1. Remove the tray cover by loosening the screws at the back of the tray.

2. Disconnect the cables from the OCP and any optional SCSI device from the bulkhead connector in the rear right corner of the tray.

3. Unscrew the Phillips head screw holding the bulkhead to the tray.

4. Unscrew the two Phillips head retaining screws and slide the tray off the drawer.

7.7 CPU Removal and Replacement

CAUTION: Several different CPU modules work in these systems. Unless you are upgrading, be sure you are replacing the broken module with the same variant. B3001 and B3002 can only be used in AlphaServer 4100 systems.

Figure 7-6   Removing CPU Module

CPU Module

System Bus
Card Cage
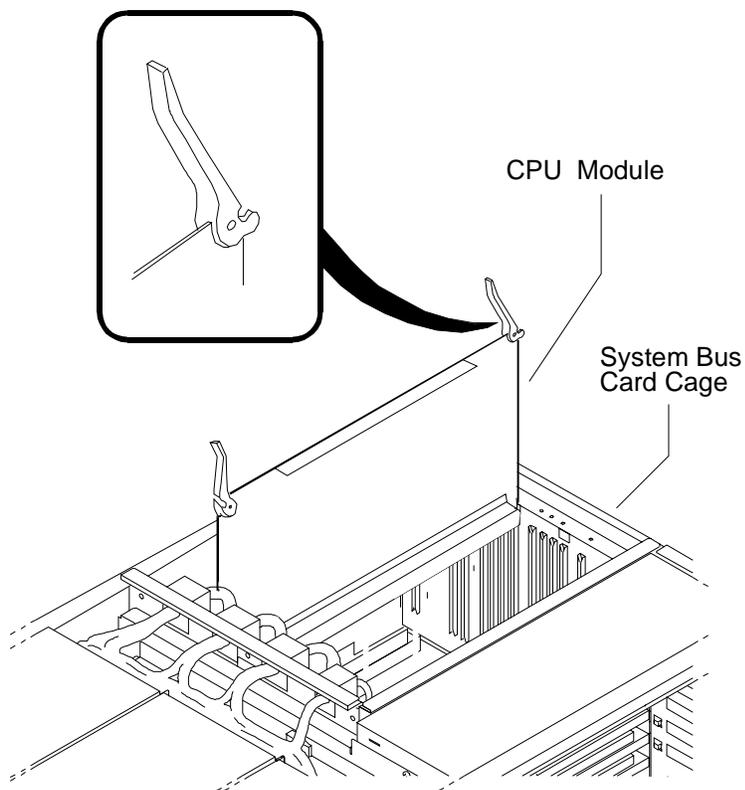
PKW0411-96

WARNING: CPU modules and memory modules have parts that operate at high temperatures. Wait 2 minutes after power is removed before touching any module.

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage. Remove the two Phillips head screws holding the cover in place and slide it off the drawer.

4. Identify and remove faulty CPU. A label to the left of the system bus card cage identifies which slot contains CPU0, CPU1, CPU2, or CPU3. The CPU is held in place with levers at both ends; simultaneously raise the levers and lift the CPU from the cage.

Replacement

Reverse the steps in the Removal procedure.

Verification — DIGITAL UNIX and OpenVMS Systems

1. Bring the system up to the SRM console by pressing the Halt button, if necessary.

2. Issue the **show cpu** command to display the status of the new module.

Verification — Windows NT Systems

1. Start **AlphaBIOS Setup**, select **Display System Configuration**, and press Enter.

2. Using the arrow keys, select **MC Bus Configuration** to display the status of the new module.

7.8 CPU Fan Removal and Replacement

Figure 7-7   Removing CPU Fan
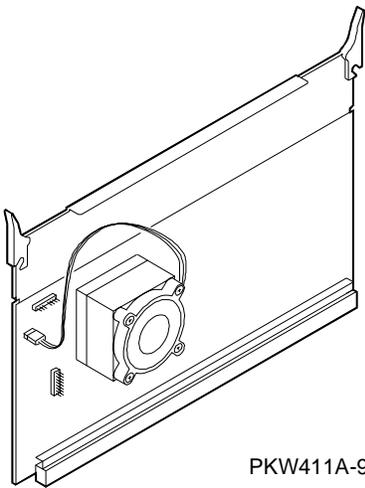
PKW411A-96

Removal

1. Follow the CPU Removal and Replacement procedure.
2. Unplug the fan from the module.
3. Remove the four Phillips head screws holding the fan to the Alpha chip's heatsink.

Replacement

Reverse the above procedure.

Verification

If the system powers up, the CPU fan is working.

7.9 Memory Removal and Replacement

CAUTION: Several different memory modules work in these systems. Be sure you are replacing the broken module with the same variant.

Figure 7-8   Removing Memory Module

Memory Module

System Bus
Card Cage
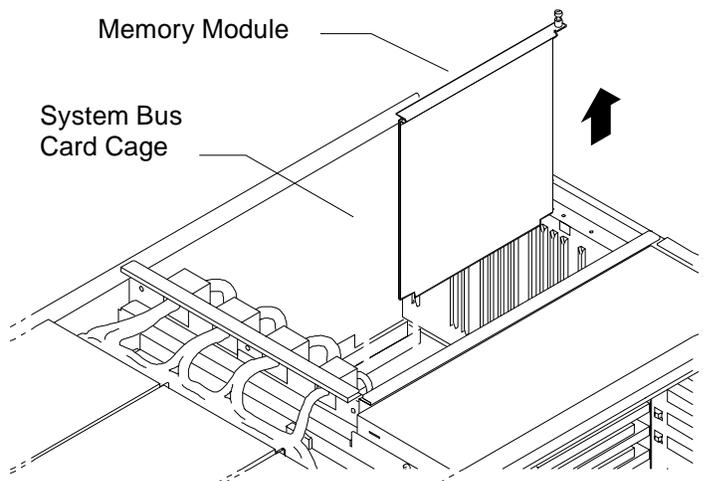
PKW0408-96

WARNING: CPU modules and memory modules have parts that operate at high temperatures. Wait 2 minutes after power is removed before touching any module.

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage. Remove the two Phillips head screws holding the cover in place and slide it off the drawer.

4. Identify and remove the faulty module. A label to the left of the system card cage identifies which slot contains the high or low halves of memory banks. The memory module is held in place by a flathead captive screw attached to the top brace of the module. Loosen the screw and lift the module from the cage.

Replacement

Reverse the steps in the Removal procedure.

NOTE: Memory modules must be installed in pairs. When you replace a bad module, be sure the second module in the pair is in place.

Verification — DIGITAL UNIX and OpenVMS Systems

1. Bring the system up to the SRM console by pressing the Halt button, if necessary.

2. Issue the **show memory** command to display the status of the new memory.

3. Verify the functioning of the new memory by issuing the command **test mem**n, where n is 0, 1, 2, 3, or *.

Verification — Windows NT Systems

1. Start **AlphaBIOS Setup**, select **Display System Configuration**, and press Enter.

2. Using the arrow keys, select **Memory Configuration** to display the status of the new memory.

3. Switch to the SRM console (press the Halt button in so that the LED on the button lights and reset the system). Verify the functioning of the new memory by issuing the command **test mem**n, where n is 0, 1, 2, 3, or *.

7.10 Power Control Module Removal and Replacement

Figure 7-9 Removing Power Control Module

Power Control
Module (PCM)

System Bus
Card Cage
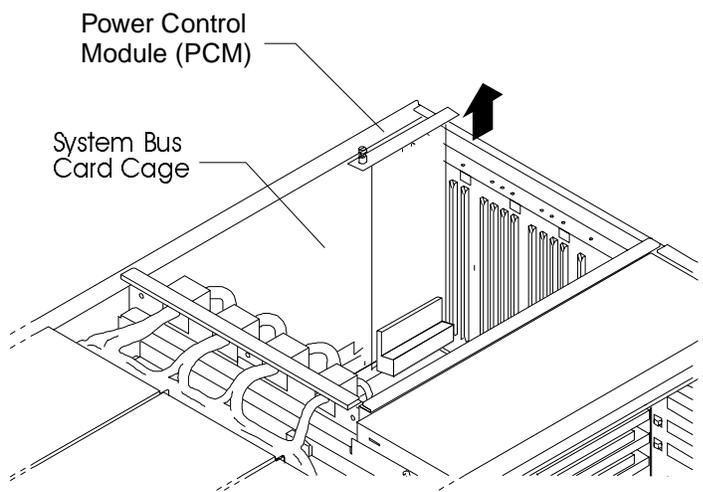
PKW0412 -96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage. Remove the two Phillips head screws holding the cover in place and slide it off the drawer.

4. Remove the faulty PCM. The PCM is located in the back left corner of the system bus card cage. A captive flathead screw and the rear card guide hold the PCM in place. Unscrew the screw and lift the module from the cage.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the PCM is faulty or not seated properly, the system will not come up.

7.11 System Bus to PCI Bus Bridge (B3040-AA) Module Removal and Replacement

Figure 7-10 Removing System Bus to PCI/EISA Bus Bridge Module (B3040-AA)
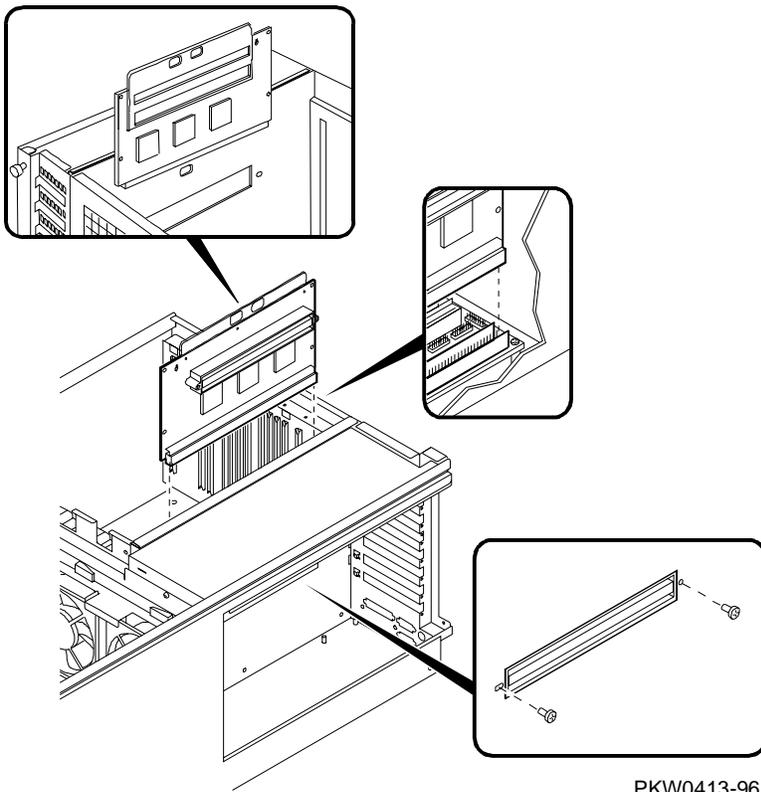
PKW0413-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage. Remove the two Phillips head screws holding the cover in place and slide it off the drawer.

4. Expose the PCI bus card cage. Remove three Phillips head screws holding the cover in place and slide it off the drawer.

5. Remove all the PCI/EISA options.

6. Remove the server control module.

7. Remove the PCI motherboard.

8. Remove the two Phillips head screws holding the system bus to PCI bus bridge module to the sheet metal between the system bus card cage and the PCI bus card cage.

9. Remove enough CPU and memory modules to the right of the bridge module to allow a flathead screwdriver to be inserted in the slot in the middle of the module's top bracket.

10. Place a flathead screwdriver into the slot in the middle of the module's top bracket and into the corresponding slot in the sheet metal between the two card cages. Use the screwdriver as a lever to disconnect the bridge module from the connector on the system motherboard.

11. Remove the bridge module from the system bus card cage.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system (press the Halt button if necessary to bring up the SRM console) and issue the **show device** command at the console prompt to verify that the system sees all system options and peripherals.

7.12 System Bus to PCI Bus Bridge (B3040-AB) Module Removal and Replacement

Figure 7-11   Removing System Bus to PCI Bus Bridge Module (B3040-AB)
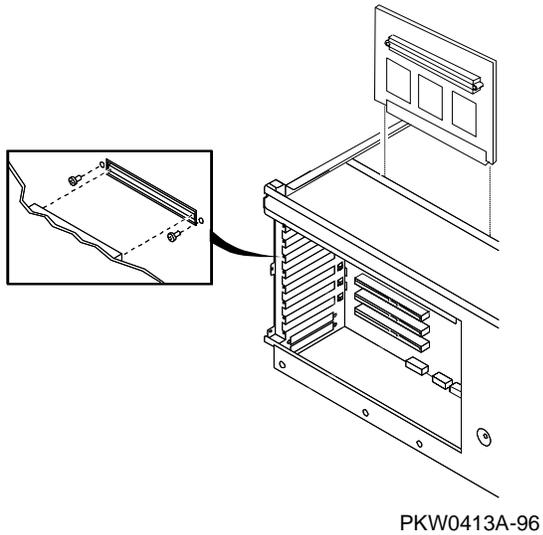
PKW0413A-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage. Remove the two Phillips head screws holding the cover in place and slide it off the drawer.

4. Expose the PCI bus card cage on the right side of the drawer. Remove three Phillips head screws holding the cover in place and slide it off the drawer.

5. Remove all the PCI options.

6. Remove the PCI motherboard.

7. Remove the two Phillips head screws holding the system bus to PCI bus bridge module to the sheet metal between the system bus card cage and the PCI bus card cage.

8. Remove enough CPU and memory modules to the left of the bridge module to allow a flathead screwdriver to be inserted in the slot in the middle of the module's top bracket.

9. Place a flathead screwdriver into the slot in the middle of the module's top bracket and into the corresponding slot in the sheet metal between the two card cages. Use the screwdriver as a lever to disconnect the bridge module from the connector on the system motherboard.

10. Remove the bridge module from the system bus card cage.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system (press the Halt button if necessary to bring up the SRM console) and issue the **show device** command at the console prompt to verify that the system sees all system options and peripherals.

7.13 System Motherboard (4100 &early 4000) Removal and Replacement

The system motherboard contains an NVRAM that holds the system serial number. Be sure to record this number before replacing the module. The serial number is on a barcode on the side of the system drawer or on the system bus card cage. The part number for the 4100 is 54-23803-01 and for the early 4000 is 54-23803-02.

Figure 7-12   Removing System Motherboard
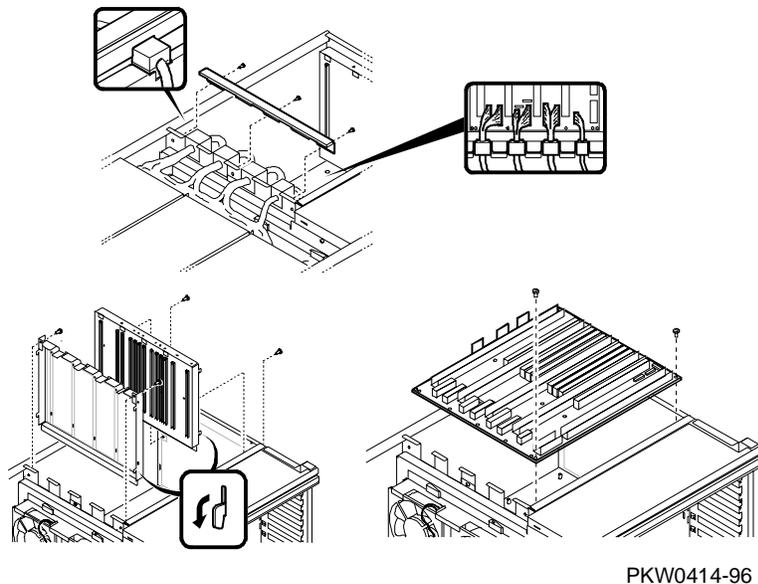
PKW0414-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage by removing the two Phillips head screws holding it in place and sliding the cover off the drawer.

4. Remove all CPUs, memory modules, and the PCM from the system motherboard.

5. Expose the PCI bus card cage. Remove three Phillips head screws holding the cover in place and slide it off the drawer.

6. Remove all the PCI/EISA options.

7. Remove the server control module.

8. Remove the PCI motherboard.

9. Remove system bus to PCI bus module from the system motherboard.

10. Remove the bracket holding the power cables in place as they pass from the system bus section to the power section of the drawer.

11. Disconnect all cables to the system motherboard and lay them back over the power supply section of the system drawer.

    CAUTION: Secure the power harness connectors in the system card cage to ensure that they cannot damage the pins in the CPU connectors.

12. Remove both the front and back module card guides. Unscrew the two screws that hold the guides in place.

13. Remove the system motherboard from the card cage by removing the 15 Phillips head screws holding it in place. Record the system serial number. (The serial number is on a barcode on the side of the system drawer or on the system bus card cage.)

Replacement

Reverse the above procedure. To align the motherboard in the cage, start replacing the screws in the corners next to the system bus to PCI bus bridge module and then the PCM module. Subsequent screws should align properly.
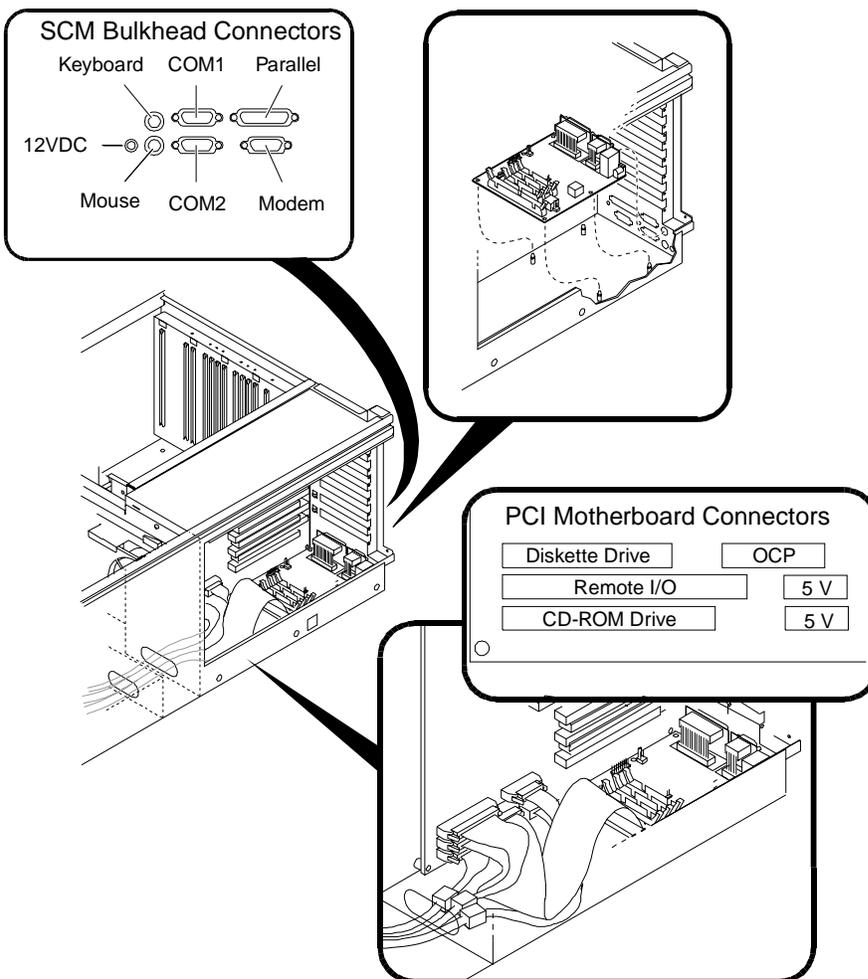
Verification

1. Power up the system (press the Halt button if necessary to bring up the SRM console) and issue the **show device** command at the console prompt to verify that all system options are seen.

2. Restore the system serial number by issuing the **set sys_serial_num** command at the SRM console prompt.

7.14 System Motherboard (4000) Removal and Replacement

The system motherboard contains an NVRAM that holds the system serial number. Be sure to record this number before replacing the module. The serial number is on a barcode on the side of the system drawer or on the system bus card cage. The part number for the later 4000 is 54-23805-01.

Figure 7-13   Removing System Motherboard
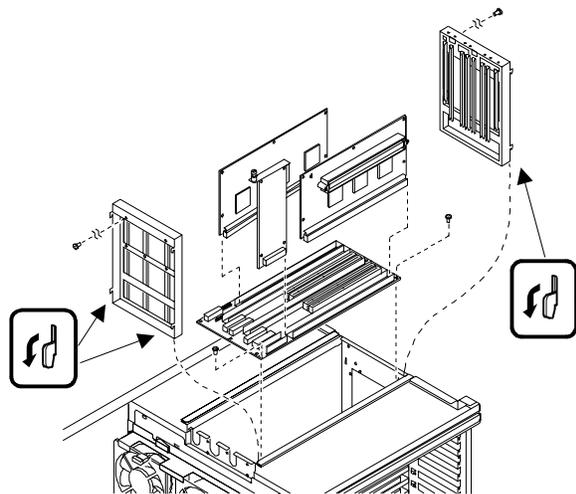
PKW0414A-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the system bus card cage by removing the two Phillips head screws holding it in place and sliding the cover off the drawer.

4. Remove all CPUs, memory modules, and the PCM from the system motherboard.

5. Expose both PCI bus card cages. Remove three Phillips head screws holding each cover in place and slide them off the drawer.

6. Remove all the PCI/EISA options.

7. Remove the server control module.

8. Remove the PCI motherboards.

9. Remove both bridge modules from the system motherboard.

10. Remove the bracket holding the power cables in place as they pass from the system bus section to the power section of the drawer.

11. Disconnect all cables to the system motherboard and lay them back over the power supply section of the system drawer.

    CAUTION: Secure the power harness connectors in the system card cage to ensure that they cannot damage the pins in the CPU connectors.

12. Remove both the front and back module card guides. Unscrew the two screws that hold the guides in place.

13. Remove the system motherboard from the card cage by removing the 15 Phillips head screws holding it in place. Record the system serial number. (The serial number is on a barcode on the side of the system drawer or on the system bus card cage.)

Replacement

Reverse the above procedure. To align the motherboard in the cage, replace screws in adjacent corners of the module. Subsequent screws should align properly.

Verification

1. Power up the system (press the Halt button if necessary to bring up the SRM console) and issue the **show device** command at the console prompt to verify that all system options are seen.

2. Restore the system serial number by issuing the **set sys_serial_num** command at the SRM console prompt.

7.15 PCI/EISA Motherboard (B3050) Removal and Replacement

Figure 7-14   Replacing PCI/EISA Motherboard

Connection to
Bridge Module

PCI Motherboard

PKW0409-96
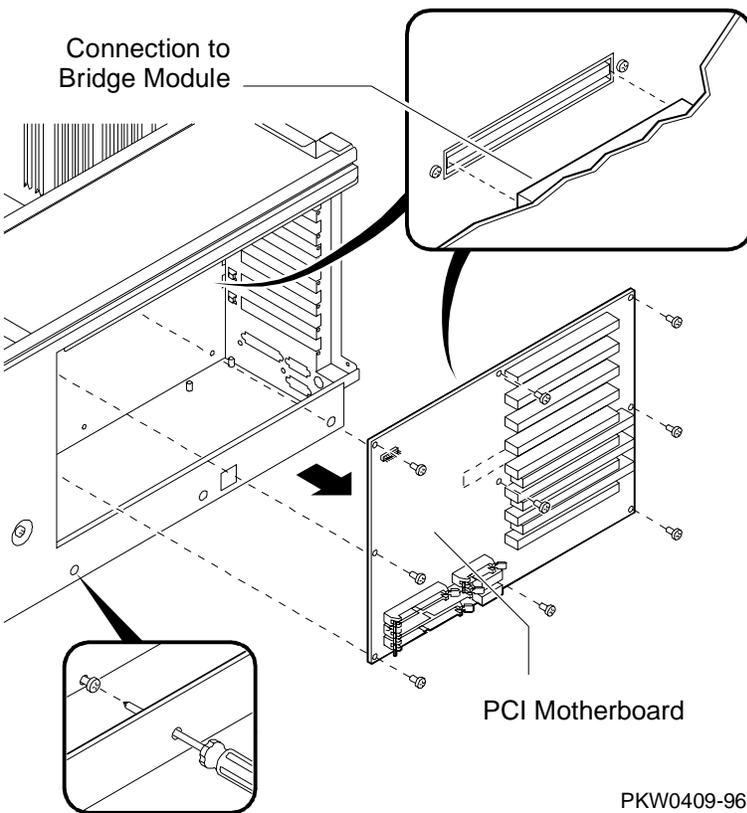
Removal

The PCI motherboard contains an NVRAM with ECU data and customized console environment variables. Therefore, if the console runs, execute a **show *** command at the console prompt and, if you have not done so earlier, record the settings for the **sys_model_number** and **sys_type** environment variables. These environment variables are used to display the system model number and type, and they compute certain information passed to the operating system. When you replace the PCI

motherboard, these environment variables are lost and must be restored after the module swap.

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the PCI bus card cage. Remove three Phillips head screws holding the cover in place and slide it off the drawer.

4. Remove all PCI and EISA options.

5. Disconnect all cables connected to the PCI motherboard.

6. Remove the server control module.

7. Unscrew the two screws holding the system bus to PCI bus bridge module in the system bus card cage to the PCI motherboard.

8. Remove the nine Phillips head screws that hold the motherboard in place. To reach the screws on the bottom of the board, thread your screwdriver through the three holes in the sheet metal.

9. Carefully pry the motherboard loose from the system bus to PCI bus bridge module on the other side of the sheet metal separating the system bus card cage from the PCI card cage.

10. Remove the motherboard from the card cage.

Replacement

Reverse the steps in the Removal procedure.

Verification

1. Power up the system (press the Halt button if necessary to bring up the SRM console) and issue the **show device** command at the console prompt to verify that the system sees all options.

2. Restore the **sys_model_num**, **sys_type,** and other customized environment variables to their previous settings. Run the ECU to restore EISA configuration data. This must be done regardless of whether there is an EISA option in the EISA slot on PCI 0.

7.16 PCI Motherboard (B3051) Removal and Replacement

Figure 7-15   Replacing PCI Motherboard
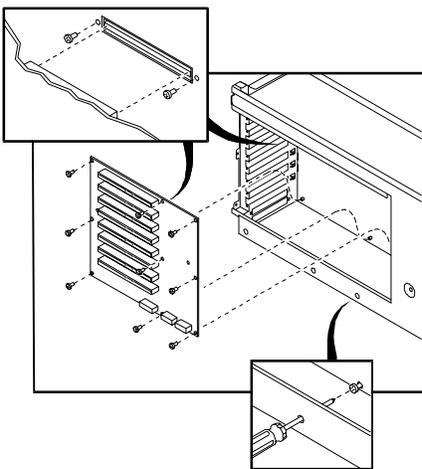
PKW0409A-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the PCI bus card cage on the right when viewing the drawer from the rear. Remove three Phillips head screws holding the cover in place, and slide it off the drawer.

4. Remove all PCI options.

5. Disconnect all cables connected to the PCI motherboard.

6. Unscrew the two screws holding the system bus to PCI bus bridge module in the system bus card cage to the PCI motherboard.

7. Remove the nine Phillips head screws that hold the motherboard in place. To reach the screws on the bottom of the board, thread your screwdriver through the three holes in the sheet metal.

8. Carefully pry the motherboard loose from the system bus to PCI bus bridge module on the other side of the sheet metal separating the system bus card cage from the PCI card cage.

9. Remove the motherboard from the card cage.

Replacement

Reverse the steps in the Removal procedure.

Verification

1. Power up the system (press the Halt button if necessary to bring up the SRM console) and issue the **show device** command at the console prompt to verify that the system sees all options.

7.17 Server Control Module Removal and Replacement

Figure 7-16 Removing Server Control Module

SCM Bulkhead Connectors

Keyboard   COM1   Parallel

12VDC

Mouse   COM2   Modem

PCI Motherboard Connectors

Diskette Drive	OCP
Remote I/O	5 V
CD-ROM Drive	5 V

PKW0415-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the PCI bus card cage. Remove three Phillips head screws holding the cover in place and slide it off the drawer.

4. Disconnect the cables connected at the bulkhead to the server control module.

5. If necessary, remove several PCI and EISA options from the bottom of the PCI card cage up until you can access the server control module.

6. Disconnect the two cables connected to the PCI motherboard at the server control module end.

7. Disconnect the twisted pair power cable from the module.

8. Place a credit car or a piece of cardboard between the edge of the SCM module and the B3050 module to protect the delicate pins of the ASICs located in the lower right corner of the B3050 module.

9. The server control module is held in place by four stud snaps. Using a flathead screwdriver gently pry the SCM module off the snaps and remove it. *Make sure you do not hit the pins of the ASICs on the B3050.*

Replacement

Reverse the steps in the Removal procedure.

Verification

Verify console output on COM1.

7.18 PCI/EISA Option Removal and Replacement

Figure 7-17   Removing PCI/EISA Option
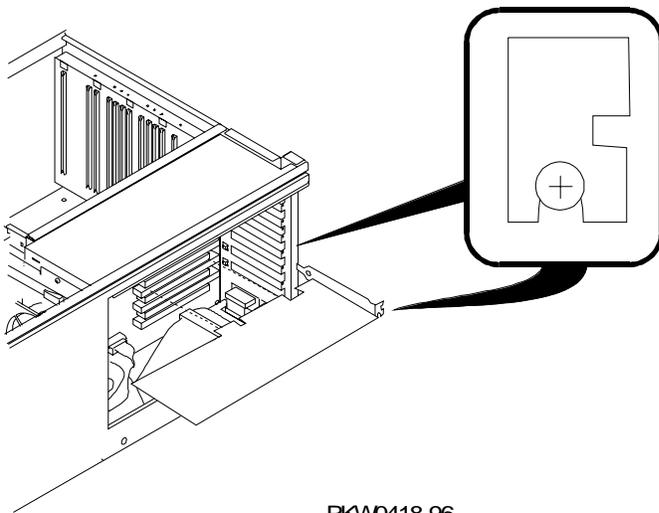
PKW0418-96

WARNING: To prevent fire, use only modules with current limited outputs. See National Electrical Code NFPA 70 or Safety of Information Technology Equipment, Including Electrical Business Equipment EN 60 950.

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the PCI bus card cage. Remove three Phillips head screws holding the cover in place and slide it off the drawer.

4. Remove the faulty option. Disconnect cables connected to the option. Unscrew the small Phillips head screw securing the option to the card cage. Slide the option from the card cage.

Replacement

Reverse the steps in the Removal procedure.

Verification — DIGITAL UNIX and OpenVMS Systems

1. Power up the system (press the Halt button if necessary to bring up the SRM console) and run the ECU to restore EISA configuration data.

2. Issue the **show config** command or **show device** command at the console prompt to verify that the system sees the option you replaced.

3. Run any diagnostic appropriate for the option you replaced.

Verification — Windows NT Systems

1. Start **AlphaBIOS Setup**, select **Display System Configuration**, and press Enter.

2. Using the arrow keys, select **PCI Configuration** or **EISA Configuration** to determine that the new option is listed.

7.19 Power Supply Removal and Replacement

Figure 7-18 Removing Power Supply
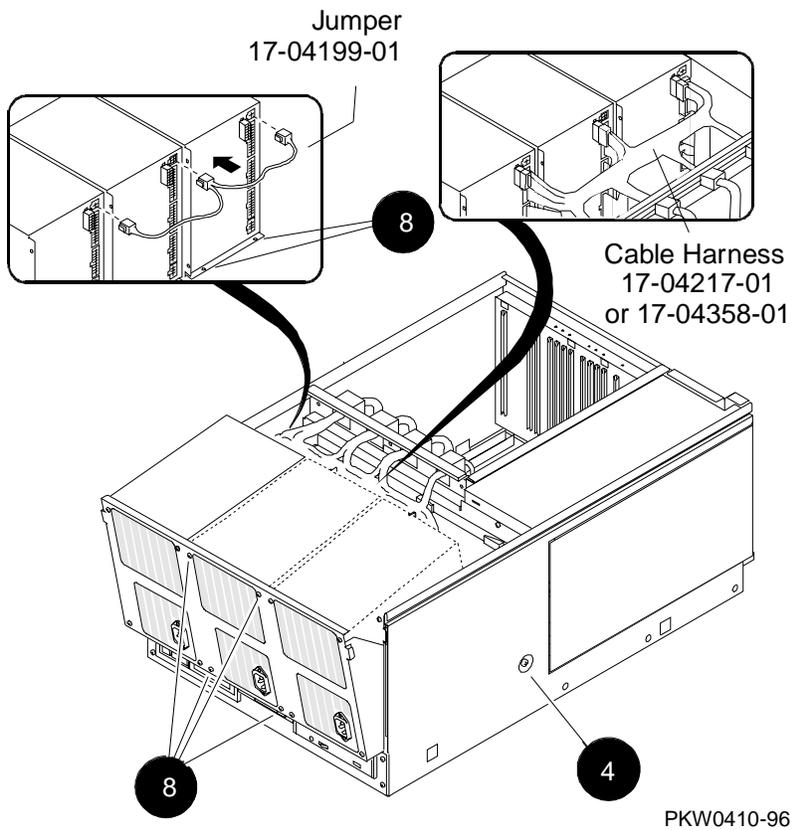
Jumper
17-04199-01

8

Cable Harness
17-04217-01
or 17-04358-01

8

4

PKW0410-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Remove the cover to the power section of the drawer. Remove the two Phillips head screws holding the cover in place and slide it off the drawer.

4. Release the power supply tray by removing the two Phillips head screws on the side of the drawer. See ❹.

5. Lift the power supply tray to release it from the sheet metal and slide it out from the drawer until it locks (about 4 inches).

6. Tilt the tray to allow easier access to the back of the power supplies.

7. Unplug the connectors at the rear of the supply that is being replaced.

8. Unscrew the four Phillips head screws at the front of the tray that hold the power supply in place. Also unscrew the two screws at the back of the power supply. See ❸.

9. Remove the power supply.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the system has redundant power, the system will power up regardless of whether the replaced power supply is faulty. In this case look at the PCM LEDs to determine that the power supply is functioning properly. If the system does not have redundant power, it will not power up.

7.20 Power Harness (4100 and early 4000) Removal and Replacement

Figure 7-19   Removing Power Harness
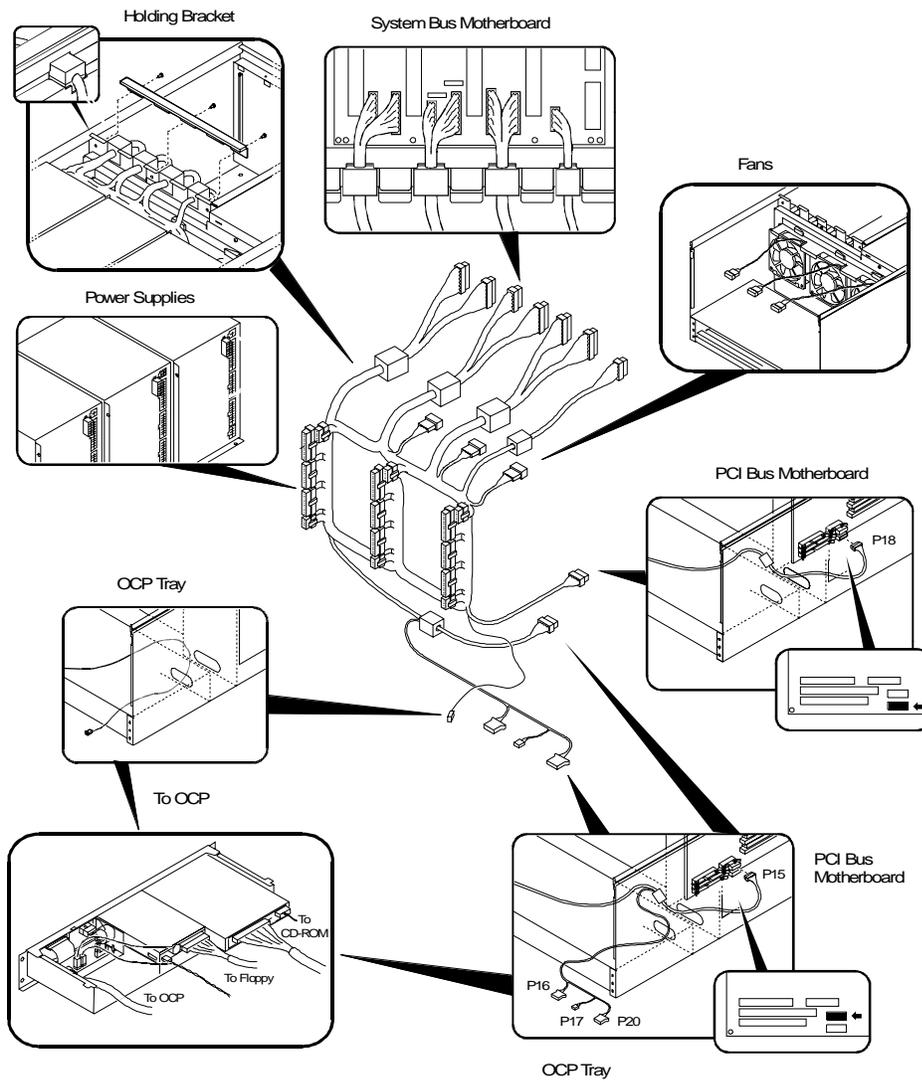
Holding Bracket

System Bus Motherboard

Fans

Power Supplies

PCI Bus Motherboard

P18

OCP Tray

To OCP

To CD-ROM

To Floppy

To OCP

PCI Bus Motherboard

P15

P16

P17   P20

OCP Tray

PKW0419-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the power, system card cage, and PCI/EISA sections of the drawer by removing all covers. Unscrew the Phillips head screws holding each cover in place and slide the covers off the drawer.

4. Release the power supply tray by removing the two Phillips head screws on the side of the drawer.

5. Lift the power supply tray to release it from the sheet metal and slide it out from the drawer until it locks.

6. Tilt the tray to allow easier access to the fans.

7. Remove the bracket holding the power harness as it passes from the power section to the system card cage section of the drawer. Remove the three Phillips head screws holding the bracket in place.

8. Disconnect the power harness from the system motherboard and fold the harness back over the power supplies.

    CAUTION: Secure the power harness connectors in the system card cage to ensure that they cannot damage the pins in the CPU connectors.

9. Disconnect the two power connectors from the PCI/ESIA motherboard. Push the power cable through the hole from the PCI/EISA section into the power section.

10. Disconnect the fan power cables from the power harness.

11. Remove the four Phillips head screws holding the OCP tray to the drawer.

12. Slide the tray from the drawer far enough to disconnect the power cables attached to the OCP (cabinet only), the floppy, and the CD-ROM.

13. As you remove the tray from the system, push the power cables through the hole at the back of the tray into the power section of the drawer.

14. Disconnect the power harness from the power supplies. Remove the harness from the system.

Replacement

Reverse the steps in the Removal procedure.

7.21 Power Harness (4000) Removal and Replacement

Figure 7-20 Removing Power Harness
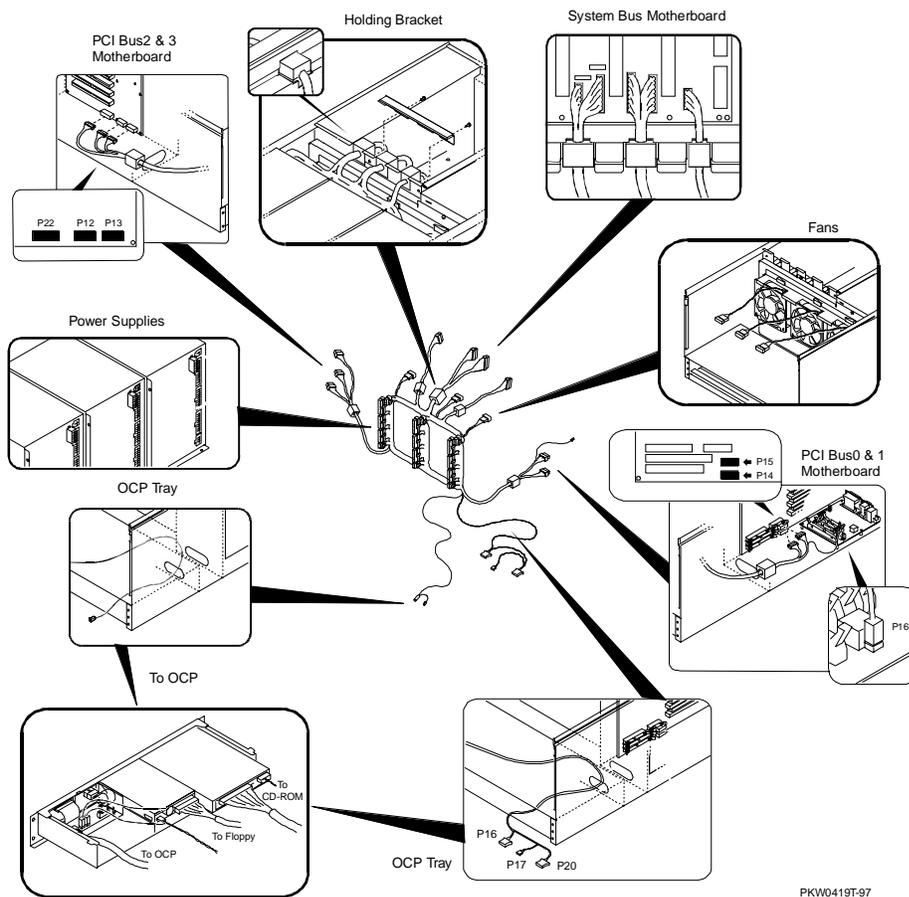
PCI Bus2 & 3 Motherboard

Holding Bracket

System Bus Motherboard

P22 P12 P13

Power Supplies

Fans

OCP Tray

PCI Bus0 & 1 Motherboard

P15
P14

P16

To OCP

To CD-ROM

To Floppy

To OCP

OCP Tray

P16

P17 P20

PKW0419T-97

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the power and system card cage sections of the drawer by removing the two covers. Unscrew the two Phillips head screws holding each cover in place and slide the covers off the drawer.

4. If you want more space to work on the fans, do this step and the next; otherwise skip to step 7. Release the power supply tray by removing the two Phillips head screws on the side of the drawer.

5. Lift the power supply tray to release it from the sheet metal and slide it out from the drawer until it locks.

6. Tilt the tray to allow easier access to the fans.

7. Remove the bracket holding the power harness as it passes from the power section to the system card cage section of the drawer. Remove the three Phillips head screws holding the bracket in place.

8. Disconnect the power harness from the system motherboard and fold the harness back over the power supplies.

   CAUTION: Secure the power harness connectors in the system card cage to ensure that they cannot damage the pins in the CPU connectors.

9. Disconnect the three power connectors from the PCI/ESIA motherboard. Push the cable through the hole from the PCI/EISA section into the power section.

10. Disconnect the three power connectors from the other PCI motherboard. Push the power cable through the hole from the PCI section into the power section.

11. Disconnect the fan power cables from the power harness.

12. Remove the four Phillips head screws holding the OCP tray to the drawer.

13. Slide the tray from the drawer far enough to disconnect the power cables attached to the OCP (cabinet only), the floppy, and the CD-ROM.

14. As you remove the tray from the system, push the power cables through the hole at the back of the tray into the power section of the drawer.

15. Disconnect the power harness from the power supplies. Remove the harness from the system.

Replacement

Reverse the steps in the Removal procedure.

7.22 System Drawer Fan Removal and Replacement

Figure 7-21   Removing System Drawer Fan
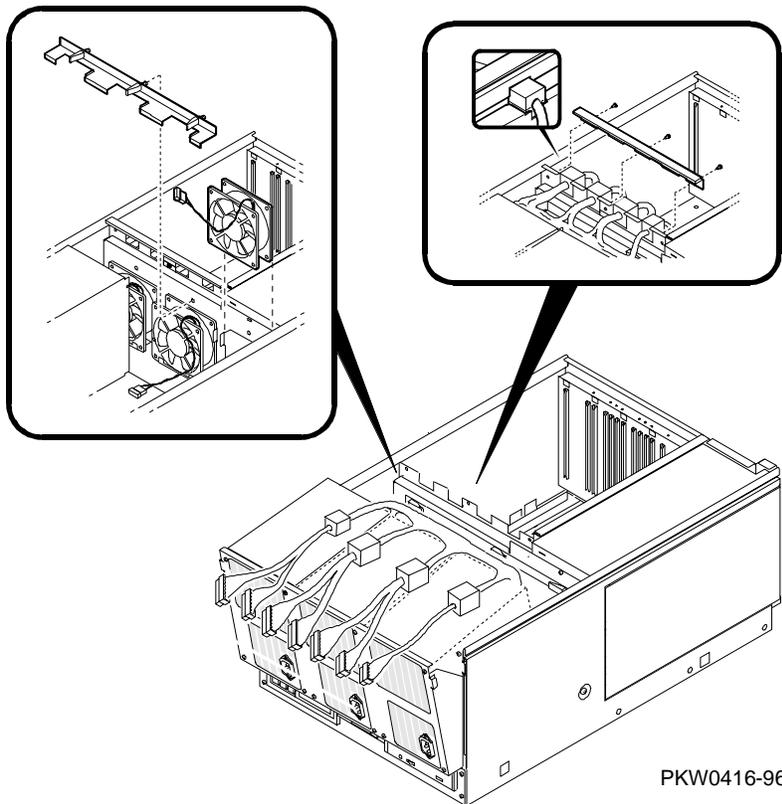
PKW0416-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Expose the power system, the system card cage, and the PCI card cage sections of the drawer by removing all three covers. Unscrew the two Phillips head screws holding each cover on top of the drawer in place and slide them off the drawer. Release the two lever latches holding the PCI card cage cover in place and slide it off.

4. Release the power supply tray by removing the two Phillips head screws on the side of the drawer.

5. Lift the power supply tray to release it from the sheet metal and slide it out from the drawer.

6. Tilt the tray to allow easier access to the fans.

7. Remove the bracket holding the power harness as it passes from the power section to the system card cage section of the drawer. Remove the three Phillips head screws holding the bracket in place.

8. Disconnect the power harness from the system motherboard and fold the harness back over the power supplies. Remove any modules that prevent you from disconnecting the harness from the system motherboard.

   CAUTION: Secure the power harness connectors in the system card cage to ensure that they cannot damage the pins in the CPU connectors.

9. Disconnect the three power connectors from the PCI motherboard and pass them through the hole from the PCI card cage to the power section of the drawer.

10. Disconnect the fan power cables from the power harness.

11. Remove the four Phillips head screws holding the OCP tray to the system drawer. Slide the tray out of the system drawer far enough to disconnect power cables attached to the OCP, the floppy, and the CD-ROM drive.

12. Remove the tray from the system.

13. Release the three lever latches on the bracket holding all three fans in place.

14. Disconnect the broken fan's power cable from the power harness and lift the fan from the drawer.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the fan you installed is faulty, the system will not power up. Look at the PCM LEDs to determine that the fan you replaced is functioning properly.

7.23 Cover Interlock (4100 and early 4000) Removal and Replacement

Figure 7-22   Removing Cover Interlocks

3 Cover
Interlock
Switches
70-32016-01

To OCP
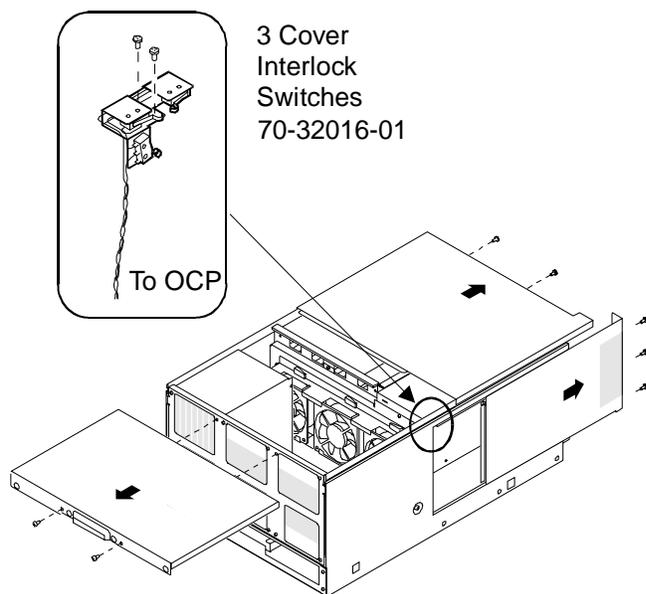
PKW-0403D-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Remove all three section covers to expose the interlock switch assembly.

4. Remove the two screws holding the interlock in place.

5. Push the interlock toward the opposite side of the system drawer (be sure not to twist it) and tilt it so that the switches affected by the power and system card cage covers clear the openings in the side of the drawer. Slide it toward the front of the drawer and remove it, letting it hang loosely over the side of the drawer.

6. If you are working on a pedestal system, disconnect the switch connection from the tray bulkhead and remove the interlock switch assembly.

7. If you are working on a system drawer in a cabinet, unscrew the four screws holding the OCP tray assembly in place beneath the drawer in front.

8. Slide the tray out and remove it from the system.

9. Pull the interlock switch connection to the OCP back through the access hole and remove the entire switch assembly.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the switch you installed is faulty, the system will not power up.

7.24 Cover Interlock (later 4000) Removal and Replacement

Figure 7-23   Removing Cover Interlocks
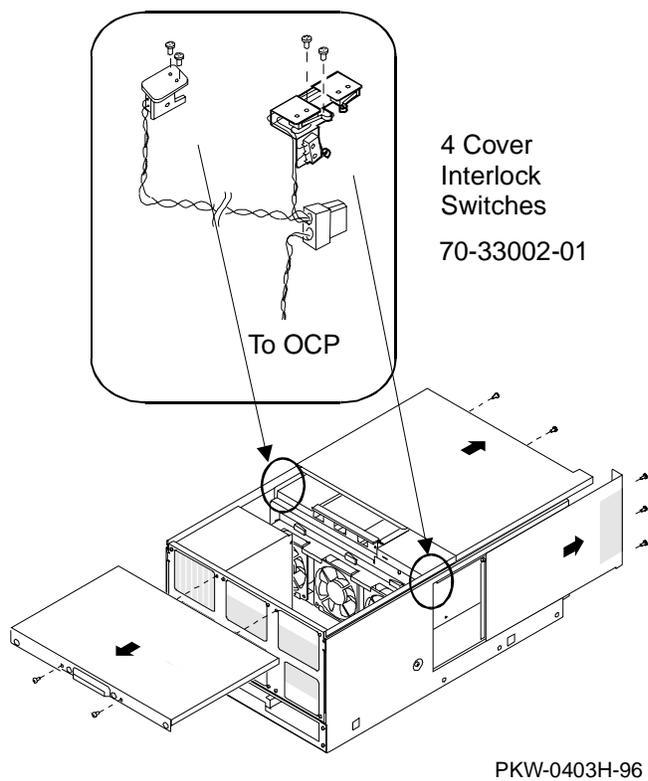
4 Cover
Interlock
Switches

70-33002-01

To OCP

PKW-0403H-96

Removal
1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Remove all three section covers to expose the interlock switch assemblies.

4. Remove the two screws holding the interlocks in place.

5. Push the interlock toward the opposite side of the system drawer (be sure not to twist it) and tilt it so that the switches affected by the power and system card cage covers clear the openings in the side of the drawer. Slide it toward the front of the drawer and remove it, letting it hang loosely over the side of the drawer.

6. If you are working on a pedestal system, disconnect the switch connection from the tray bulkhead and remove the interlock switch assembly.

7. If you are working on a system drawer in a cabinet, unscrew the four screws holding the OCP tray assembly in place beneath the drawer in front.

8. Slide the tray out and remove it from the system.

9. Pull the interlock switch connection to the OCP back through the access hole and remove the entire switch assembly.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the switch you installed is faulty, the system will not power up.

7.25 Operator Control Panel Removal and Replacement (Cabinet)

Figure 7-24   Removing OCP (Cabinet)
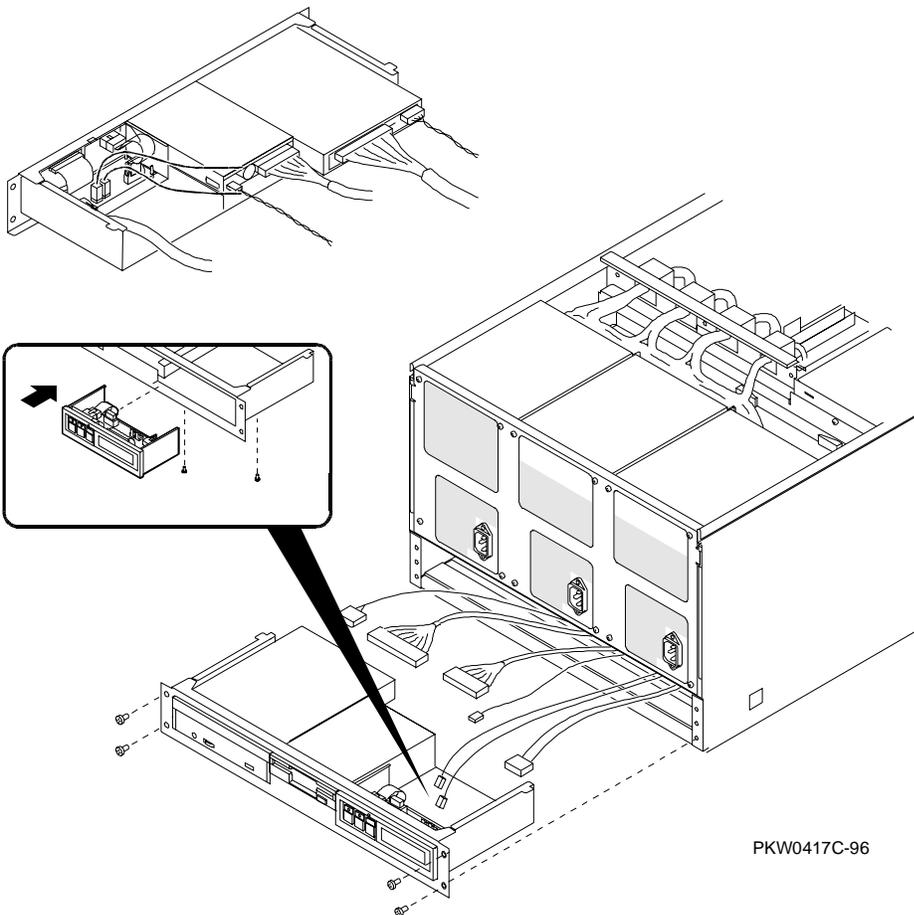
PKW0417C-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. While you need not remove the tray containing the OCP, you do need to slide it forward to access the OCP retaining screws under the tray. The tray is attached to the power system section cover. To slide the tray forward:

   a. Remove the tray cover by loosening the retaining screws at the back of the tray and sliding it toward the back of the system.

   b. Disconnect the cables from the OCP, and any optional SCSI device in the tray from the bulkhead at the rear right of the tray.

   c. Unscrew the Phillips head retaining screw holding the bulkhead to the tray.

   d. Unscrew the two Phillips head retaining screws at the front of the system drawer and slide the tray forward.

4. Remove the white power interconnect wire and the signal ribbon cable from the OCP.

5. Remove the two Phillips head screws holding the OCP in place and remove it from the tray.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the OCP you installed is faulty, the system will not power up.

7.26 Operator Control Panel Removal and Replacement (Pedestal)

Figure 7-25   Removing OCP (Pedestal)
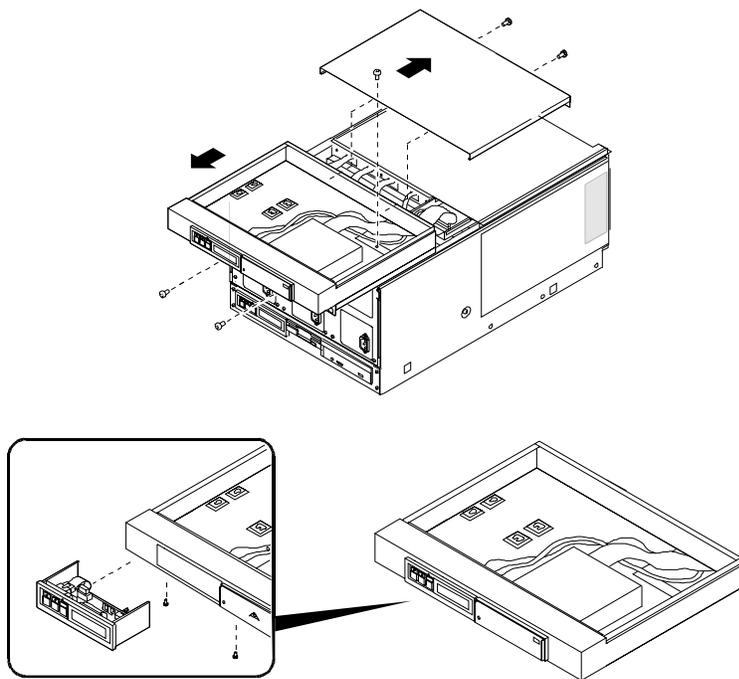
PKW0430 -96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Remove the four Phillips head screws holding the OCP tray to the system drawer.

4. Slide the tray out of the system drawer far enough to disconnect cables attached to the OCP, the floppy, and the CD-ROM drive.

5. Remove the tray from the system.

6. Move the tray to some handy work surface. Hold the tray vertically and remove the two Phillips head screws that hold the OCP in place from the bottom of the tray and remove the OCP assembly from the tray.

Replacement

Reverse the steps in the Removal procedure. As you replace the tray in the drawer, be sure that the slides on the sides of the tray are placed on the rails in the drawer.

Verification

Power up the system. If the OCP you installed is faulty, the system will not power up or you will not see messages on the OCP display.

7.27 Floppy Removal and Replacement

Figure 7-26   Removing Floppy Drive
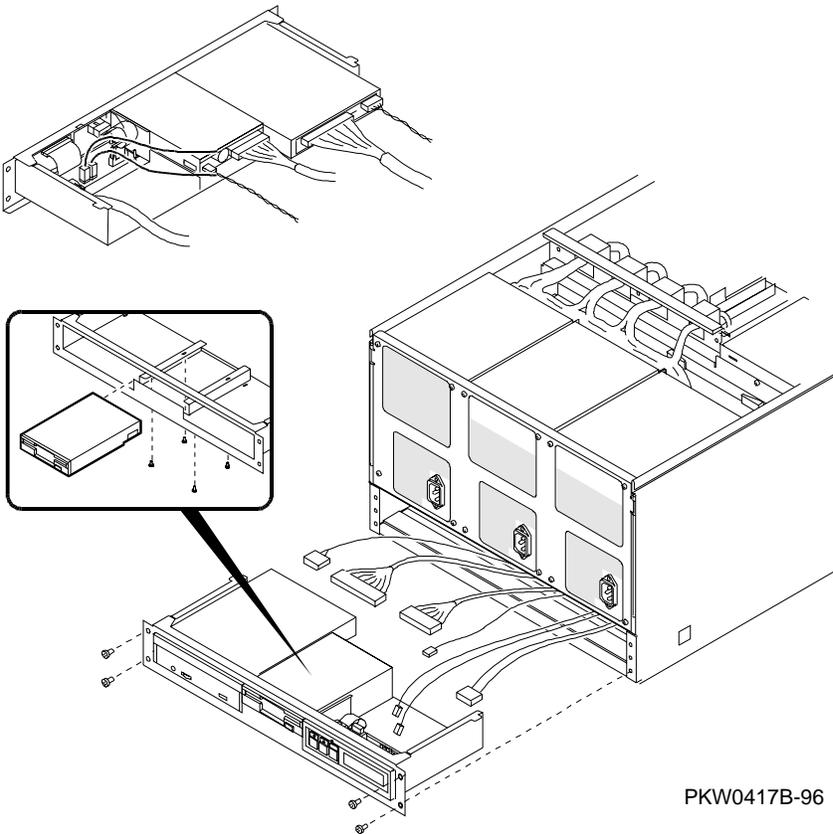
PKW0417B-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Remove the four Phillips head screws holding the OCP tray to the system drawer.

4. Slide the tray out of the system drawer and disconnect cables attached to the OCP (unnecessary on a pedestal system), the floppy, and the CD-ROM drive. (In the pedestal system the OCP is in the tray above the power supplies.)

5. Move the tray to some handy work surface. Hold the tray vertically and from the bottom of the tray remove the four Phillips head screws that hold the floppy in place and remove it from the tray.

Replacement

Reverse the steps in the Removal procedure. As you replace the tray in the drawer, be sure that the slides on the sides of the tray are placed on the rails in the drawer.

Verification

Power up the system. Use the following SRM console commands to test the floppy:

```
P00>>> show dev floppy
P00>>> HD buf/dva0
```

7.28 CD-ROM Removal and Replacement

Figure 7-27 Removing CD-ROM
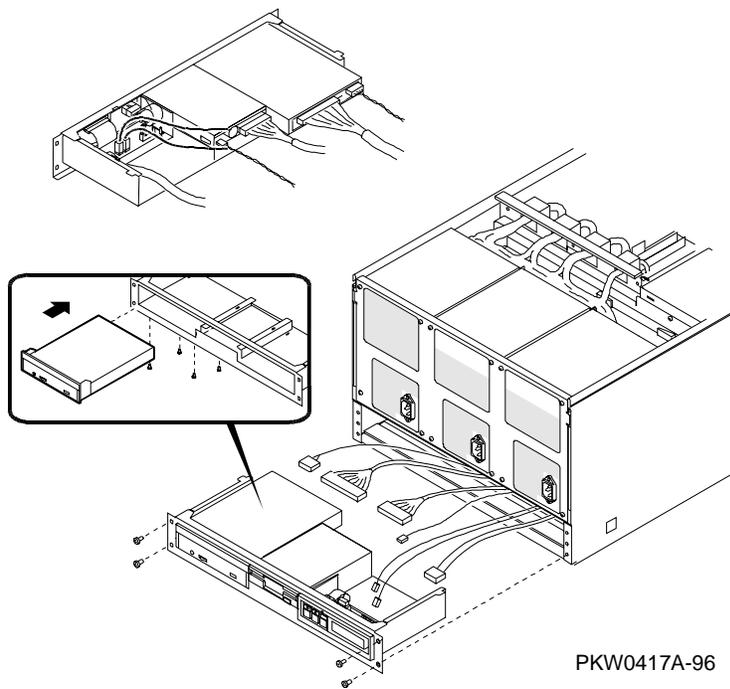
PKW0417A-96

Removal

1. Shut down the operating system and power down the system.

2. Expose the system drawer.

3. Remove the four Phillips head screws holding the OCP tray to the system drawer.

4. Slide the tray out of the system drawer and disconnect cables attached to the OCP (unnecessary on a pedestal system), the floppy, and the CD-ROM drive. (In the pedestal system the OCP is in the pedestal tray above the power supplies.)

5. Move the tray to some handy work surface. Hold the tray vertically and from the bottom of the tray remove the four Phillips head screws that hold the floppy in place and remove it from the tray.

Replacement

Reverse the steps in the Removal procedure. As you replace the tray in the drawer, be sure that the slides on the sides of the tray are placed on the rails in the drawer.

Verification

Power up the system (press the Halt button if necessary to bring up the SRM console). Use the following SRM console commands to test the CD-ROM:

```
P00>>> show dev ncr0
P00>>> HD buf/dka nnn
```
where *nnn* is the device number; for example, dka500.

7.29 Cabinet Fan Tray Removal and Replacement

Figure 7-28 Removing Cabinet Fan Tray

Fan LED
Power LED

AC
Power

To SCM

PKW0441A-96
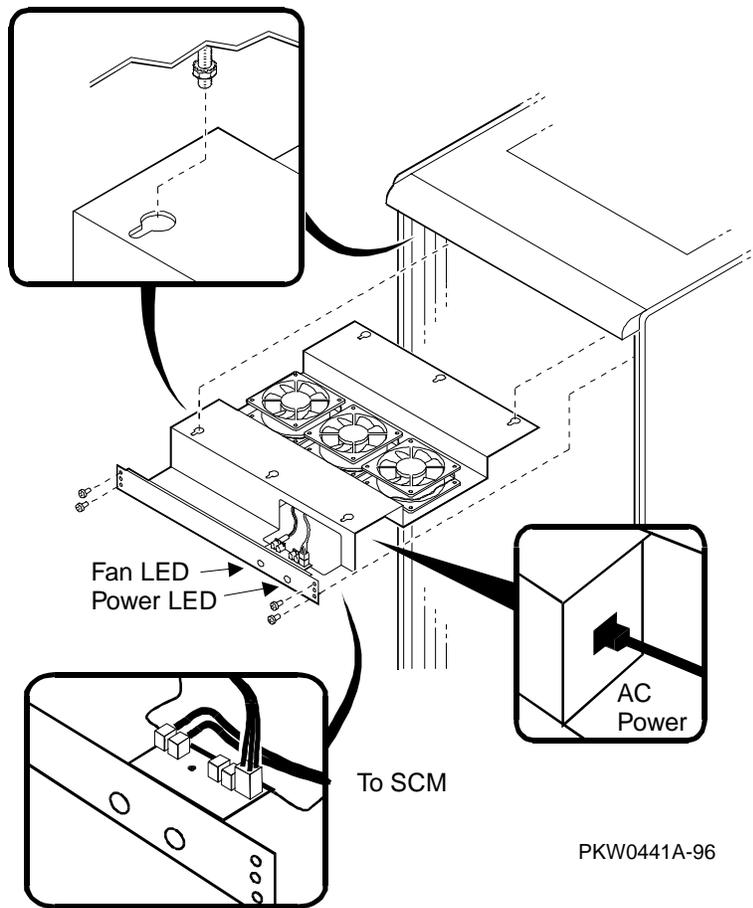
Removal

1. Shut down the operating system and power down the system. Unplug the AC power cable from the cabinet tray power supply.

2. If present, unplug any power cables going to the server control modules at the back of system drawers.

3. Unscrew the four Phillips head screws securing the fan tray to the top of the cabinet.

4. Loosen the four hexnuts that hold the tray to the top of the cabinet.

5. Holding the bottom of the tray, slide it out so that the holes in the tray frame slip over the loosened hexnuts.

6. Move the tray to a work surface to remove whatever component is being replaced.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. If the green power LED comes on, and the fan LED is off, the cabinet fan tray is verified.

7.30 Cabinet Fan Tray Power Supply Removal and Replacement

Figure 7-29   Removing Cabinet Fan Tray Power Supply
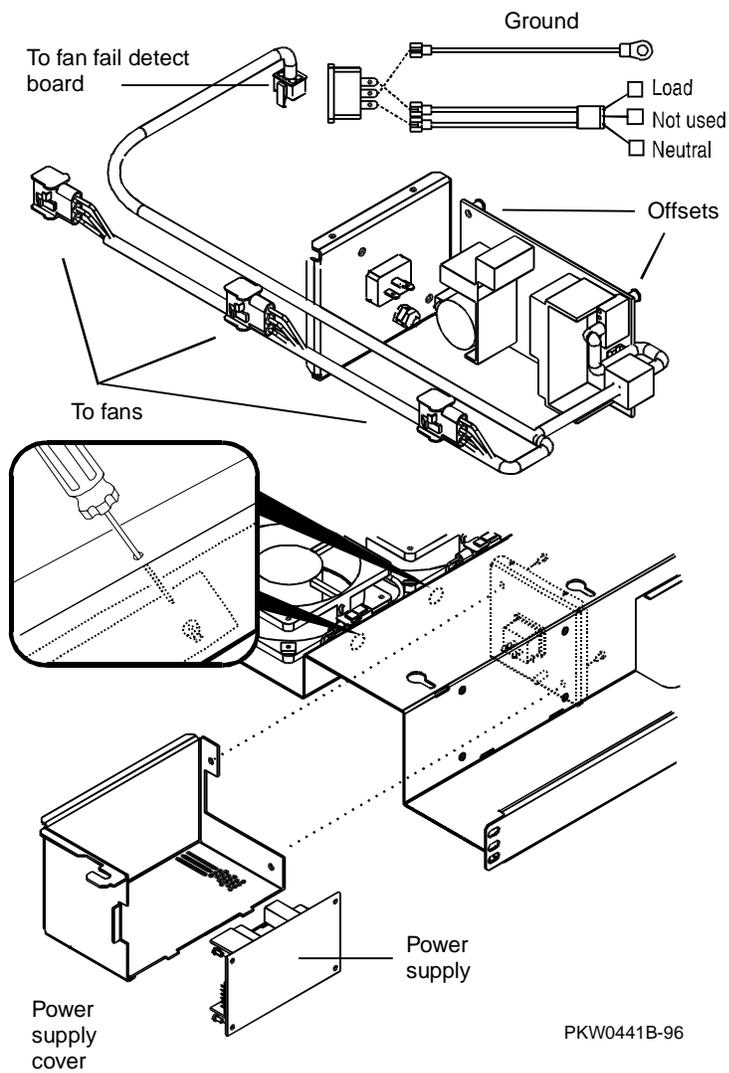
PKW0441B-96

Removal

1. Remove the cabinet fan tray.

2. Disconnect the power harness from the fan fail detect module and each fan.

3. Remove the power supply cover. It is held in place by two screws that go through the AC bulkhead spot welded to the tray weldment.

4. Remove the power harness from the tray by disconnecting it from the power supply.

5. Disconnect the neutral and load leads from the power supply.

6. Remove the four screws holding the power supply to the tray. Keep track of the standoffs that provide space between the power supply and weldment. You will need them during replacement.

Replacement

1. Reverse the steps in the Removal procedure.

2. Place the fan tray back in the cabinet.

Verification

Power up the system. If the green power LED comes on, and the fan LED is off, the cabinet fan tray power supply is verified.

7.31 Cabinet Fan Tray Fan Removal and Replacement

Figure 7-30   Removing Cabinet Fan Tray Fan
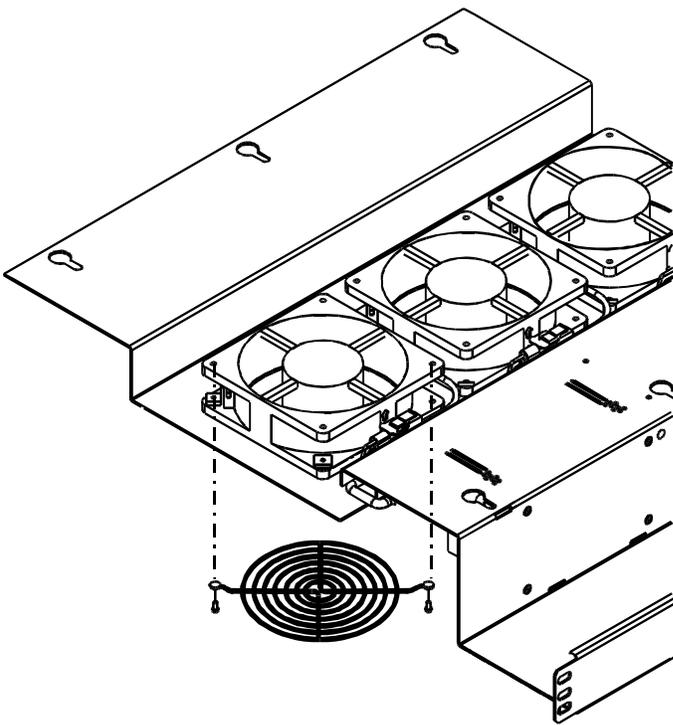
PKW0441F-96

Removal

1. Remove the cabinet fan tray.

2. Disconnect the power harness from the fan you wish to replace.

3. Remove the fan finger guard.

4. Remove the two remaining screws holding the fan to the tray and remove the fan.

5. If the new fan does not have clip nuts, remove them from the fan.

Replacement

1. Reverse the Removal procedure, taking care to orient the fan so that the connection to the power harness is dressed nicely.

2. Place the fan tray back in the cabinet.

Verification

Power up the system. If the green power LED comes on, and the fan LED is off, the cabinet fan tray fan is verified.

7.32 Cabinet Fan Tray Fan Fail Detect Module Removal and Replacement

Figure 7-31   Removing Fan Tray Fan Fail Detect Module
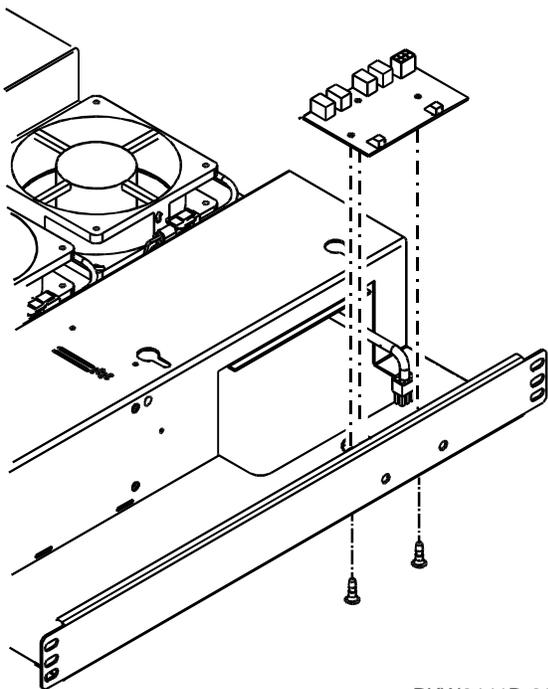
PKW0441D-96

Removal

1. Remove the cabinet fan tray.

2. Disconnect the power harness from the fan fail detect module.

3. Remove the fan fail detect module. In early systems, the module is held in place by three screws that go through the weldment, through three standoffs, through the module to nuts. In later systems, the module snaps in place.

Replacement

1. Reverse the steps in the Removal procedure.

2. Place the fan tray back in the cabinet.

Verification

Power up the system. If the green power LED comes on, and the fan LED is off, the cabinet fan fail detect module is verified.

7.33 StorageWorks Shelf Removal and Replacement

Figure 7-32   Removing StorageWorks Shelf

Cabinet

StorageWorks Shelf
Mounting Rails
(H910A-EC)

StorageWorks Shelf
Mounting Rails
(H910A-EB)

Pedestal
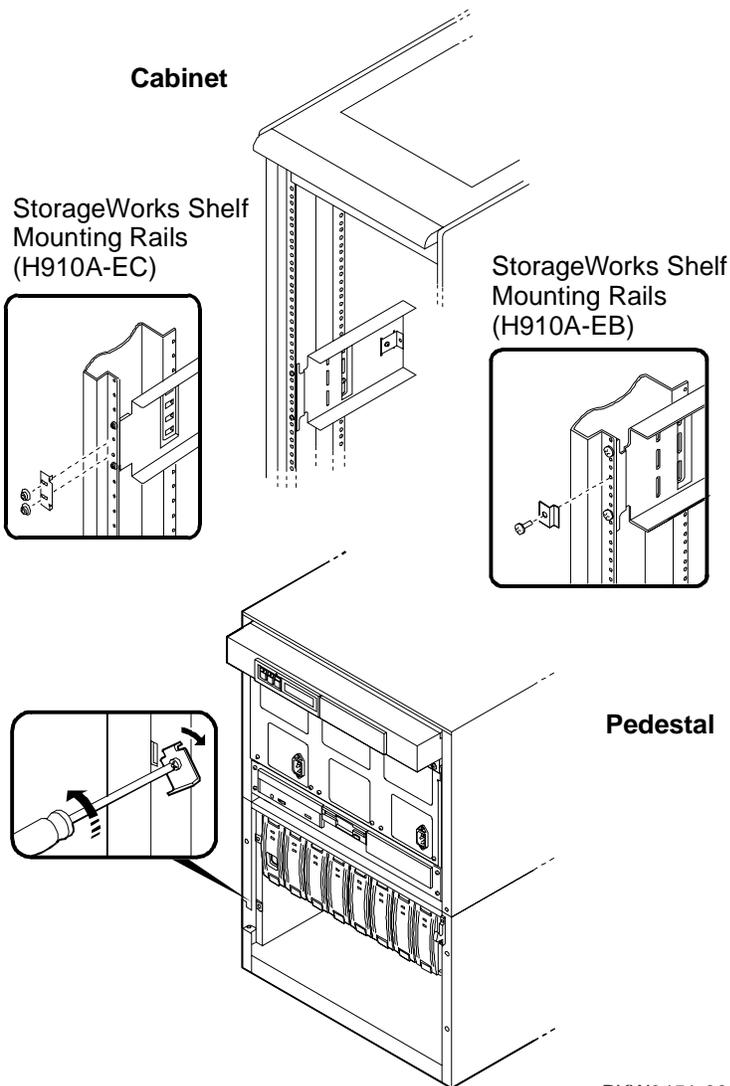
PKW0451-96

Removal

1. Shut down the operating system and power down the system.

2. Remove the power cord and signal cord(s) from the StorageWorks shelf.

3. Remove the two retaining brackets holding the shelf in the mounting rail by removing the Phillips head screws holding the brackets in place.

4. Slide the shelf out of the system.

Replacement

Reverse the steps in the Removal procedure.

Verification

Power up the system. Use the **show device console** command to verify that the StorageWorks shelf is configured into the system.

Appendix A
Running Utilities

This appendix provides a brief overview of how to load and run utilities. The following topics are covered:

- Running Utilities from a Graphics Monitor
- Running Utilities from a Serial Terminal
- Running ECU
- Running RAID Standalone Configuration Utility
- Updating Firmware with LFU
- Updating Firmware from AlphaBIOS
- Upgrading AlphaBIOS

A.1 Running Utilities from a Graphics Monitor

Start AlphaBIOS and select Utilities from the menu. The next selection depends on the utility to be run. For example, to run ECU, select Run ECU from floppy. To run RCU, select Run Maintenance Program.

Figure A-1 Running a Utility from a Graphics Monitor

```
                            AlphaBIOS Setup              F1=Help

     Display System Configuration...
     Upgrade AlphaBIOS
     Hard Disk Setup...
     CMOS Setup...
     Install Windows NT
     Utilities                      ▶ Run ECU from floppy...
     About AlphaBIOS...               OS Selection Setup...
                                      Run Maintenance Program...
```
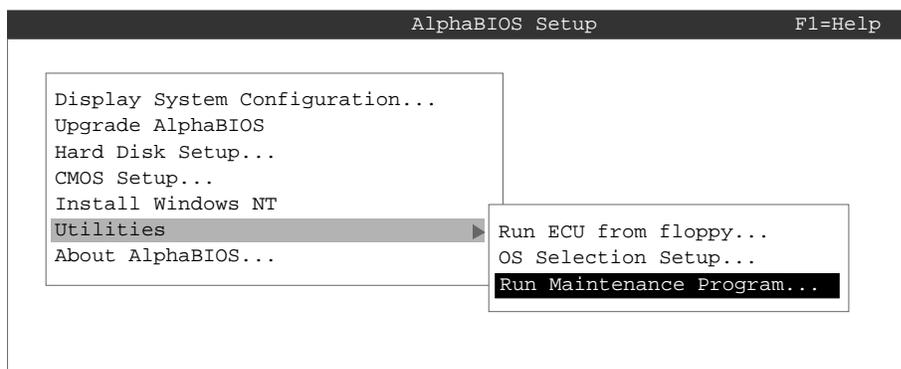
PK-0729-96

A.2 Running Utilities from a Serial Terminal

Utilities are run from a serial terminal in the same way as from a graphics monitor. The menus are the same, but some keys are different.

Table A-1 AlphaBIOS Option Key Mapping

AlphaBIOS Key	VTxxx Key
F1	Ctrl/A
F2	Ctrl/B
F3	Ctrl/C
F4	Ctrl/D
F5	Ctrl/E
F6	Ctrl/F
F7	Ctrl/P
F8	Ctrl/R
F9	Ctrl/T
F10	Ctrl/U
Insert	Ctrl/V
Delete	Ctrl/W
Backspace	Ctrl/H
Escape	Ctrl/[

A.3   Running ECU

The EISA Configuration Utility (ECU) is used to configure EISA options on AlphaServer systems. The ECU can be run either from a graphics monitor or a serial terminal.

1. Start AlphaBIOS Setup. If the system is in the SRM console, issue the command **alphabios**. (If the system has a graphics monitor, you can set the SRM **console** environment variable to **graphics**.)

2. From AlphaBIOS Setup, select Utilities, then select Run ECU from floppy… from the submenu that displays, and press Enter.

    NOTE: The EISA Configuration Utility is supplied on diskettes shipped with the system. There is a diskette for Microsoft Windows NT and a diskette for DIGITAL UNIX and OpenVMS.

3. Insert the correct ECU diskette for the operating system and press Enter to run it.

The ECU main menu displays the following options:

```
EISA Configuration Utility
Steps in configuring your computer

STEP 1:  Important EISA configuration information
STEP 2:  Add or remove boards
STEP 3:  View or edit details
STEP 4:  Examine required details
STEP 5:  Save and exit
```

NOTE: Step 1 of the ECU provides online help. It is recommended that you select this step and become familiar with the utility before proceeding.

A.4 Running RAID Standalone Configuration Utility

The RAID Standalone Configuration Utility is used to set up RAID disk drives and logical units. The Standalone Utility is run from the AlphaBIOS Utility menu.

The AlphaServer 4100 system supports the KZPSC-*xx* PCI RAID controller (SWXCR). The KZPSC-*xx* kit includes the controller, RAID Array 230 Subsystems software, and documentation.

1. Start AlphaBIOS Setup. If the system is in the SRM console, issue the command **alphabios**. (If the system has a graphics monitor, you can set the SRM **console** environment variable to **graphics**.)

2. At the Utilities screen, select Run Maintenance Program. Press Enter.

3. In the Run Maintenance Program dialog box, type **swxcrmgr** in the Program Name: field.

4. Press Enter to execute the program. The Main menu displays the following options:

```
[01.View/Update Configuration]
 02.Automatic Configuration
 03.New Configuration
 04.Initialize Logical Drive
 05.Parity Check
 06.Rebuild
 07.Tools
 08.Select SWXCR
 09.Controller Setup
 10.Diagnostics
```

Refer to the RAID Array Subsystems documentation for information on using the Standalone Configuration Utility to set up RAID drives.

A.5 Updating Firmware with LFU

Start the Loadable Firmware Update (LFU) utility by issuing the lfu command at the SRM console prompt or by selecting Update AlphaBIOS in the AlphaBIOS Setup screen. LFU is part of the SRM console.

Example A-1 Starting LFU from the SRM Console

```
P00>>> lfu

          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: cda0
.
.
.
UPD>
```

Figure A-2 Starting LFU from the AlphaBIOS Console

```
┌──────────────────────────────────────────────────────────┐
│                      AlphaBIOS Setup                       │
├──────────────────────────────────────────────────────────┤
│                                                            │
│    ┌──────────────────────────────────────┐               │
│    │ Display System Configuration...      │               │
│    │ Upgrade AlphaBIOS                    │               │
│    │ Hard Disk Setup                      │               │
│    │ CMOS Setup...                        │               │
│    │ Install Windows NT                   │               │
│    │ Utilities                        ►   │               │
│    │ About AlphaBIOS...                   │               │
│    └──────────────────────────────────────┘               │
│                                                            │
│    ┌──────────────────────────────────────────────────┐   │
│    │ Press ENTER to upgrade your AlphaBIOS from floppy  │   │
│    │ or CD-ROM.                                         │   │
│    │                                                    │   │
│    └──────────────────────────────────────────────────┘   │
│                                                            │
│  ESC=Exit                                                  │
└──────────────────────────────────────────────────────────┘
```
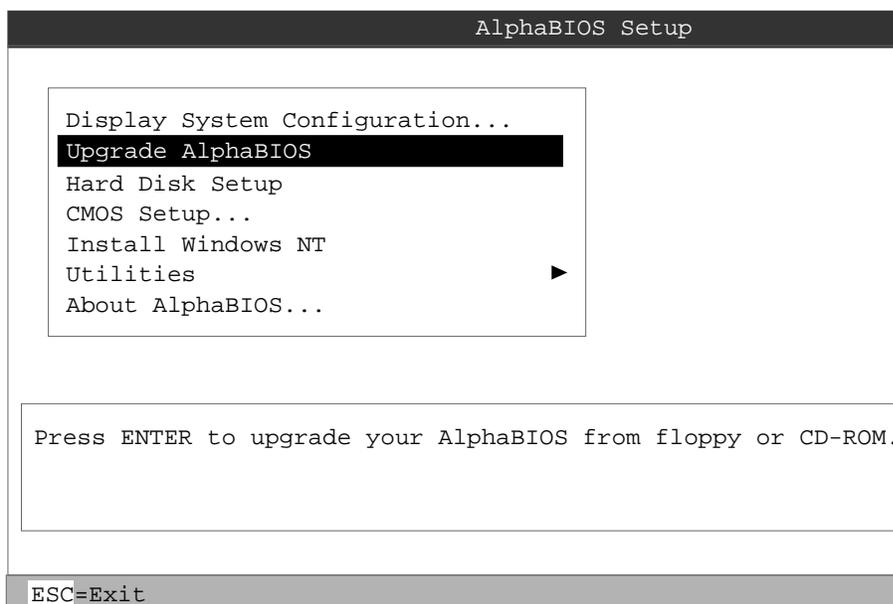
PK-0726A-96

Use the Loadable Firmware Update (LFU) utility to update system firmware. You can start LFU from either the SRM console or the AlphaBIOS console.

- From the SRM console, start LFU by issuing the **lfu** command.

- From the AlphaBIOS console, select **Upgrade AlphaBIOS** from the **AlphaBIOS Setup** screen (see Figure A-2).

A typical update procedure is:

1. Start LFU.

2. Use the LFU **list** command to show the revisions of modules that LFU can update and the revisions of update firmware.

3. Use the LFU **update** command to write the new firmware.

4. Use the LFU **exit** command to exit back to the console.

The sections that follow show examples of updating firmware from the local CD-ROM, the local floppy, and a network device. Following the examples is an LFU command reference.

A.5.1   Updating Firmware from the Internal CD-ROM

Insert the update CD-ROM, start LFU, and select cda0 as the load device.

Example A-2   Updating Firmware from the Internal CD-ROM

```
          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: cda0          ❶

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS4X00FW]: AS4X00CP ❷

Copying AS4X00CP from DKA500.5.0.1.1 .
Copying [as4x00]RHREADME from DKA500.5.0.1.1 .
Copying [as4x00]RHSRMROM from DKA500.5.0.1.1 ....................
Copying [as4x00]RHARCROM from DKA500.5.0.1.1 .............
-----------------------------------------------------------------
 Function    Description                                          ❸
-----------------------------------------------------------------
 Display     Displays the system's configuration table.
 Exit        Done exit LFU (reset).
 List        Lists the device, revision, firmware name, and
             update revision.
 Lfu         Restarts LFU.
 Readme      Lists important release information.
 Update      Replaces current firmware with loadable data image.
 Verify      Compares loadable and hardware images.
 ? or Help   Scrolls this function table.
-----------------------------------------------------------------
UPD> list                                                        ❹

Device          Current Revision    Filename      Update Revision
AlphaBIOS         V5.12-2           arcrom           V6.40-1
srmflash          V1.0-9           srmrom           V2.0-3
```
Continued on next page

❶ Select the device from which firmware will be loaded. The choices are the internal CD-ROM, the internal floppy disk, or a network device. In this example, the internal CD-ROM is selected.

❷ Select the file that has the firmware update, or press Enter to select the default file. The file options are:

AS4X00FW (default)   SRM console, AlphaBIOS console, and I/O adapter firmware

AS4X00CP   SRM console and AlphaBIOS console firmware only

AS4X00IO   I/O adapter firmware only

In this example the file for console firmware (AlphaBIOS and SRM) is selected.

❸ The LFU function table and prompt (UPD>) display.

❹ Use the LFU **list** command to determine the revision of firmware in a device and the most recent revision of that firmware available in the selected file. In this example, the resident firmware for each console (SRM and AlphaBIOS) is at an earlier revision than the firmware in the update file.

Example A-2   Updating Firmware from the Internal CD-ROM (Continued)

```
UPD> update *                                                    ❺
WARNING: updates may take several minutes to complete for each
device.

Confirm update on: AlphaBIOS        [Y/(N)] y                    ❻

                    DO NOT ABORT!
AlphaBIOS     Updating to V6.40-1... Verifying V6.40-1... PASSED.

Confirm update on: srmflash         [Y/(N)] y

                    DO NOT ABORT!
srmflash      Updating to V2.0-3...  Verifying V2.0-3...  PASSED.

UPD> exit                                                        ❼
```

❺   The **update** command updates the device specified or all devices. In this
    example, the wildcard indicates that all devices supported by the selected
    update file will be updated.

❻   For each device, you are asked to confirm that you want to update the
    firmware. The default is no. Once the update begins, do not abort the
    operation. Doing so will corrupt the firmware on the module.

❼   The **exit** command returns you to the console from which you entered LFU
    (either SRM or AlphaBIOS).

A.5.2 Updating Firmware from the Internal Floppy Disk — Creating the Diskettes

Create the update diskettes before starting LFU. See Section A.5.3 for an example of the update procedure.

Table A-2 File Locations for Creating Update Diskettes on a PC

Console Update Diskette	I/O Update Diskette
AS4X00FW.TXT	AS4X00IO.TXT
AS4X00CP.TXT	RHREADME.SYS
RHREADME.SYS	CIPCA214.SYS
RHSRMROM.SYS	DFPAA246.SYS
RHARCROM.SYS	KZPAAA10.SYS

To update system firmware from floppy disk, you first must create the firmware update diskettes. You will need to create two diskettes: one for console updates, and one for I/O.

1. Download the update files from the Internet (see the Preface of this book).

2. On a PC, copy files onto two FAT-formatted diskettes.

   From an OpenVMS system, copy files onto two ODS2-formatted diskettes as shown in Example A-3.

Example A-3   Creating Update Diskettes on an OpenVMS System

Console Update Diskette

```
$ inquire ignore "Insert blank HD floppy in DVA0, then continue"
$ set verify
$ set proc/priv=all
$ init /density=hd/index=begin dva0: rhods2cp
$ mount dva0: rhods2cp
$ create /directory dva0:[as4x00]
$ copy as4x00fw.sys dva0:[as4x00]as4x00fw.sys
$ copy as4x00cp.sys dva0:[as4x00]as4x00cp.sys
$ copy rhreadme.sys dva0:[as4x00]rhreadme.sys
$ copy as4x00fw.txt dva0:[as4x00]as4x00fw.txt
$ copy as4x00cp.txt dva0:[as4x00]as4x00cp.txt
$ copy rhsrmrom.sys dva0:[as4x00]rhsrmrom.sys
$ copy rharcrom.sys dva0:[as4x00]rharcrom.sys
$ dismount dva0:
$ set noverify
$ exit
```

I/O Update Diskette

```
$ inquire ignore "Insert blank HD floppy in DVA0, then continue"
$ set verify
$ set proc/priv=all
$ init /density=hd/index=begin dva0: rhods2io
$ mount dva0: rhods2io
$ create /directory dva0:[as4x00]
$ create /directory dva0:[options]
$ copy as4x00fw.sys dva0:[as4x00]as4x00fw.sys
$ copy as4x00io.sys dva0:[as4x00]as4x00io.sys
$ copy rhreadme.sys dva0:[as4x00]rhreadme.sys
$ copy as4x00fw.txt dva0:[as4x00]as4x00fw.txt
$ copy as4x00io.txt dva0:[as4x00]as4x00io.txt
$ copy cipca214.sys dva0:[options]cipca214.sys
$ copy dfpaa246.sys dva0:[options]dfpaa246.sys
$ copy kzpsaA10.sys dva0:[options]kzpsaa10.sys
$ dismount dva0:
$ set noverify
$ exit
```

A.5.3 Updating Firmware from the Internal Floppy Disk — Performing the Update

Insert an update diskette (see Section A.5.2) into the internal floppy drive. Start LFU and select dva0 as the load device.

Example A-4 Updating Firmware from the Internal Floppy Disk

```
            ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: dva0          ❶

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS4X00IO,(AS4X00CP)]:
AS4X00IO                                                          ❷

Copying AS4X00IO from DVA0 .
Copying RHREADME from DVA0 .
Copying CIPCA214 from DVA0 .
Copying DFPAA252 from DVA0 ...
Copying KZPSAA11 from DVA0 ...
.
.  (The function table displays, followed by the UPD> prompt, as
.   shown in Example A-2.)

UPD> list                                                         ❸

Device          Current Revision    Filename      Update Revision
AlphaBIOS         V5.12-3           arcrom        Missing file
pfi0               2.46             dfpaa_fw         2.52
srmflash          T3.2-21           srmrom        Missing file
                                    cipca_fw        A214
                                    kzpsa_fw         A11
```

Continued on next page

❶   Select the device from which firmware will be loaded. The choices are the internal CD-ROM, the internal floppy disk, or a network device. In this example, the internal floppy disk is selected.

❷   Select the file that has the firmware update, or press Enter to select the default file. When the internal floppy disk is the load device, the file options are:

AS4X00CP (default)   SRM console and AlphaBIOS console firmware only

AS4X00IO   I/O adapter firmware only

The default option in Example A-2 (AS4X00FW) is not available, since the file is too large to fit on a 1.44 MB diskette. This means that when a floppy disk is the load device, you can update either console firmware or I/O adapter firmware, but not both in the same LFU session. If you need to update both, after finishing the first update, restart LFU with the **lfu** command and insert the floppy disk with the other file.

In this example the file for I/O adapter firmware is selected.

❸   Use the LFU **list** command to determine the revision of firmware in a device and the most recent revision of that firmware available in the selected file. In this example, the update revision for console firmware displays as "Missing file" because only the I/O firmware files are available on the floppy disk.

Continued on next page

Example A-4 Updating Firmware from the Internal Floppy Disk(Continued)

```
UPD> update pfi0                                              ❹
WARNING: updates may take several minutes to complete for each
device.

Confirm update on: pfi0                 [Y/(N)] y             ❺

                        DO NOT ABORT!
pfi0            Updating to 2.52... Verifying to 2.52... PASSED.

UPD> lfu                                                      ❻

        ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: dva0

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS4X00IO,(AS4X00CP)]:❼


.
. (The function table displays, followed by the UPD> prompt.
.  Console firmware can now be updated.)

UPD> exit                                                    ❽
```

❹   The **update** command updates the device specified or all devices.

❺   For each device, you are asked to confirm that you want to update the
      firmware. The default is no. Once the update begins, do not abort the
      operation. Doing so will corrupt the firmware on the module.

❻   The **lfu** command restarts the utility so that console firmware can be updated.
      (Another method is shown in Example A-5, where the user specifies the file
      AS4X00FW and is prompted to insert the second diskette.)

❼   The default update file, AS4X00CP, is selected. The console firmware can now
      be updated, using the same procedure as for the I/O firmware.

❽   The **exit** command returns you to the console from which you entered LFU
      (either SRM or AlphaBIOS).

Example A-5 Selecting AS4X00FW to Update Firmware from the Internal Floppy Disk

```
P00>>> lfu

          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: dva0

Please enter the name of the firmware files list, or
Press <return> to use the default filename [AS4X00IO,(AS4X00CP)]: as4x00fw

Copying AS4X00FW from DVA0 .
Copying RHREADME from DVA0 .
Copying RHSRMROM from DVA0 ........................
Copying RHARCROM from DVA0 ..............
Copying CIPCA214 from DVA0
Please insert next floppy containing the firmware,
Press <return> when ready. Or type DONE to abort.
Copying CIPCA214 from DVA0 .
Copying DFPAA246 from DVA0 ...
Copying KZPSAA10 from DVA0 ...
.
.
.
```

A.5.4 Updating Firmware from a Network Device

Copy files to the local MOP server's MOP load area, start LFU, and select ewa0 as the load device.

Example A-6 Updating Firmware from a Network Device

```
          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: ewa0        ❶

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS4X00FW]:           ❷

Copying AS4X00FW from EWA0 .
Copying RHREADME from EWA0 .
Copying RHSRMROM from EWA0 .........................
Copying RHARCROM from EWA0 ............
Copying CIPCA214 from EWA0 .
Copying DFPAA246 from EWA0 ...
Copying KZPSAA11 from EWA0 ...
```

.
. [The function table displays, followed by the UPD> prompt, as
. shown in Example A-2.]

```
UPD> list                                                       ❸

Device            Current Revision   Filename      Update Revision
AlphaBIOS         V5.12-2            arcrom          V6.40-1
kzpsa0            A10                kzpsa_fw        A11
kzpsa1            A10                kzpsa_fw        A11
srmflash          V1.0-9             srmrom          V2.0-3
                                     cipca_fw        A214
                                     dfpaa_fw         2.46
```

Continued on next page

Before starting LFU, download the update files from the Internet (see Preface). You will need the files with the extension .SYS. Copy these files to your local MOP server's MOP load area.

❶   Select the device from which firmware will be loaded. The choices are the internal CD-ROM, the internal floppy disk, or a network device. In this example, a network device is selected.

❷   Select the file that has the firmware update, or press Enter to select the default file. The file options are:

AS4X00FW   SRM console, AlphaBIOS console, and I/O adapter
(default)   firmware

AS4X00CP   SRM console and AlphaBIOS console firmware only

AS4X00IO   I/O adapter firmware only

In this example the default file, which has both console firmware (AlphaBIOS and SRM) and I/O adapter firmware, is selected.

❸   Use the LFU **list** command to determine the revision of firmware in a device and the most recent revision of that firmware available in the selected file. In this example, the resident firmware for each console (SRM and AlphaBIOS) and I/O adapter is at an earlier revision than the firmware in the update file.

Example A-6 Updating Firmware from a Network Device (Continued)

```
UPD> update * -all                                              ❹
WARNING: updates may take several minutes to complete for each
device.

                         DO NOT ABORT!
AlphaBIOS    Updating to V6.40-1... Verifying V6.40-1...  PASSED.

                         DO NOT ABORT!
kzpsa0       Updating to  A11    ...  Verifying A11...  PASSED.

                         DO NOT ABORT!
kzpsa1       Updating to  A11    ...  Verifying A11...  PASSED.

                         DO NOT ABORT!
srmflash     Updating to V2.0-3...  Verifying V2.0-3...  PASSED.

UPD> exit                                                      ❺
```

❹ The **update** command updates the device specified or all devices. In this example, the wildcard indicates that all devices supported by the selected update file will be updated. Typically, LFU requests confirmation before updating each console's or device's firmware. The **-all** option removes the update confirmation requests.

❺ The **exit** command returns you to the console from which you entered LFU (either SRM or AlphaBIOS).

A.5.5   LFU Commands

The commands summarized in Table A-3 are used to update system firmware.

Table A-3   LFU Command Summary

Command	Function
display	Shows the system physical configuration.
exit	Terminates the LFU program.
help	Displays the LFU command list.
lfu	Restarts the LFU program.
list	Displays the inventory of update firmware on the selected device.
readme	Lists release notes for the LFU program.
update	Writes new firmware to the module.
verify	Reads the firmware from the module into memory and compares it with the update firmware.

These commands are described in the following pages.

display

The **display** command shows the system physical configuration. **Display** is equivalent to issuing the SRM console command **show configuration**. Because it shows the slot for each module, **display** can help you identify the location of a device.

exit

The **exit** command terminates the LFU program, causes system initialization and testing, and returns the system to the console from which LFU was called.

help

The **help** (or **?**) command displays the LFU command list, shown below.

```
--------------------------------------------------------------------
 Function    Description
--------------------------------------------------------------------
 Display     Displays the system's configuration table.
 Exit        Done exit LFU (reset).
 List        Lists the device, revision, firmware name, and update
             revision.
 Lfu         Restarts LFU.
 Readme      Lists important release information.
 Update      Replaces current firmware with loadable data image.
 Verify      Compares loadable and hardware images.
 ? or Help   Scrolls this function table.
--------------------------------------------------------------------
```

lfu

The **lfu** command restarts the LFU program. This command is used when the update files are on a floppy disk. The files for updating both console firmware and I/O firmware are too large to fit on a 1.44 MB disk, so only one type of firmware can be updated at a time. Restarting LFU enables you to specify another update file.

list

The **list** command displays the inventory of update firmware on the CD-ROM, network, or floppy. Only the devices listed at your terminal are supported for firmware updates.

The **list** command shows three pieces of information for each device:

- Current Revision — The revision of the device's current firmware
- Filename — The name of the file used to update that firmware
- Update revision — The revision of the firmware update image

readme

The **readme** command lists release notes for the LFU program.

update

The **update** command writes new firmware to the module. Then LFU automatically verifies the update by reading the new firmware image from the module into memory and comparing it with the source image.

To update more than one device, you may use a wildcard but not a list. For example, **update k*** updates all devices with names beginning with k, and **update *** updates all devices. When you do not specify a device name, LFU tries to update all devices; it lists the selected devices to update and prompts before devices are updated. (The default is no.) The **-all** option removes the update confirmation requests, enabling the update to proceed without operator intervention.

*CAUTION: Never abort an **update** operation. Aborting corrupts the firmware on the module.*

verify

The **verify** command reads the firmware from the module into memory and compares it with the update firmware. If a module already verified successfully when you updated it, but later failed tests, you can use **verify** to tell whether the firmware has become corrupted.

A.6 Updating Firmware from AlphaBIOS

Insert the CD-ROM or diskette with the updated firmware and select Upgrade AlphaBIOS from the main AlphaBIOS Setup screen. Use the Loadable Firmware Update (LFU) utility to perform the update. The LFU exit command causes a system reset.

Figure A-3   AlphaBIOS Setup Screen

```
                         AlphaBIOS Setup

     ┌──────────────────────────────────────────────┐
     │   Display System Configuration...            │
     │   Upgrade AlphaBIOS                          │
     │   Hard Disk Setup                            │
     │   CMOS Setup...                              │
     │   Install Windows NT                         │
     │   Utilities                         ▶        │
     │   About AlphaBIOS...                         │
     └──────────────────────────────────────────────┘


   ┌────────────────────────────────────────────────────┐
   │ Press ENTER to upgrade your AlphaBIOS from floppy or CD-ROM. │
   │                                                      │
   │                                                      │
   └────────────────────────────────────────────────────┘

   ESC=Exit
```
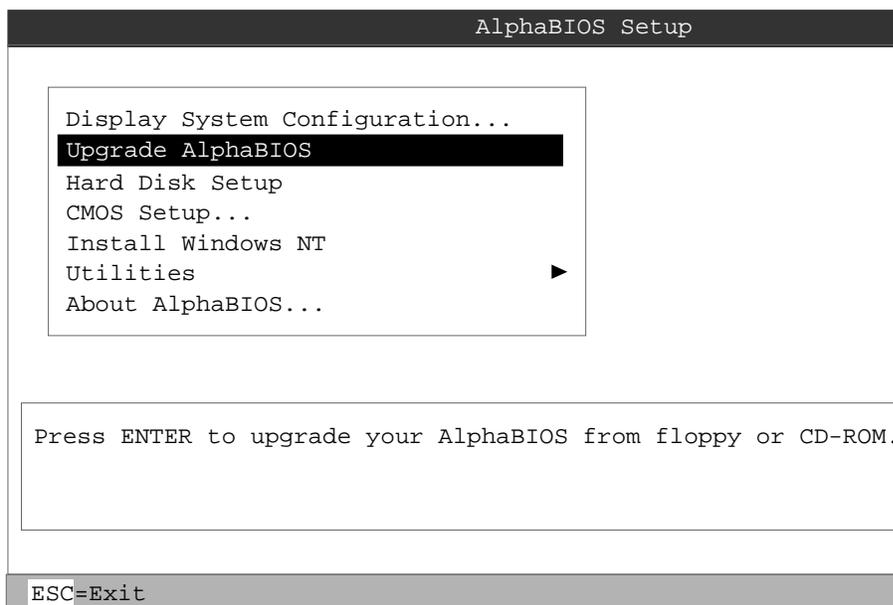
PK-0726A-96

Upgrading AlphaBIOS

As new versions of Windows NT are released, it might be necessary to upgrade AlphaBIOS to the latest version. Additionally, as improvements are made to AlphaBIOS, it might be desirable to upgrade to take advantage of new AlphaBIOS features.

Use this procedure to upgrade from an earlier version of AlphaBIOS:

1. Insert the diskette or CD-ROM containing the AlphaBIOS upgrade.

2. If you are not already running **AlphaBIOS Setup**, start it by restarting your system and pressing F2 when the **Boot** screen is displayed.

3. In the main **AlphaBIOS Setup** screen, select **Upgrade AlphaBIOS** and press Enter.

   The system is reset and the Loadable Firmware Update (LFU) utility is started. See Section A5.5 for LFU commands.

4. When the upgrade is complete, issue the LFU **exit** command. The system is reset and you are returned to AlphaBIOS.

   If you press the Reset button instead of issuing the LFU **exit** command, the system is reset and you are returned to LFU.

Appendix B

SRM Console Commands and Environment Variables

This appendix provides a summary of the SRM console commands and environment variables. The **test** command is described in Chapter 3 of this document. For complete reference information on the other SRM commands and environment variables, see the *AlphaServer 4000/4100 System Drawer User's Guide*.

NOTE: It is recommended that you keep a list of the environment variable settings for systems that you service, because you will need to restore certain environment variable settings after swapping modules. Refer to Table B-3 for a convenient worksheet.

B.1 Summary of SRM Console Commands

The SRM console commands are used to examine or modify the system state.

Table B-1   Summary of SRM Console Commands

Command	Function
alphabios	Loads and starts the AlphaBIOS console.
boot	Loads and starts the operating system.
clear *envar*	Resets an environment variable to its default value.
continue	Resumes program execution.
crash	Forces a crash dump at the operating system level.
deposit	Writes data to the specified address.
edit	Invokes the console line editor on a RAM file or on the nvram file (power-up script).
examine	Displays the contents of a memory location, register, or device.
halt	Halts the specified processor. (Same as **stop**.)
help	Displays information about the specified console command.
info *num*	Displays various types of information about the system:
	Info shows a list describing the **num** qualifier.
	Info 3 reads the impure area that contains the state of the CPU before it entered PAL mode.
	Info 5 reads the PAL built logout area that contains the data used by the operating system to create the error entry
	Info 8 reads the IOD and IOD1 registers.
initialize	Resets the system.
lfu	Runs the Loadable Firmware Update Utility.

Continued on next page

Table B-1 Summary of SRM Console Commands (Continued)

Command	Function
man	Displays information about the specified console command.
more	Displays a file one screen at a time.
prcache	Initializes and displays status of the PCI NVRAM.
set *envar*	Sets or modifies the value of an environment variable.
set host	Connects to an MSCP DUP server on a DSSI device.
set rcm_dialout	Sets a modem dialout string.
show *envar*	Displays the state of the specified environment variable.
show config	Displays the configuration at the last system initialization.
show cpu	Displays the state of each processor in the system.
show device	Displays a list of controllers and their devices in the system.
show fru	Displays the serial number and revision level of system bus options.
show memory	Displays memory module information.
show network	Displays the state of network devices in the system.
show pal	Displays the version of the privileged architecture library code (PALcode).
show power	Displays information about the power supplies, system fans, CPU fans, and temperature.
show rcm_dialout	Displays the modem dialout string.
show version	Displays the version of the console program.
start	Starts a program that was previously loaded on the processor specified.
stop	Halts the specified processor. (Same as **halt**.)
test	Runs firmware diagnostics for the system.

B.2 Summary of SRM Environment Variables

Environment variables pass configuration information between the console and the operating system. Their settings determine how the system powers up, boots the operating system, and operates. Environment variables are set or changed with the set *envar* command and returned to their default values with the clear *envar* command. Their values are viewed with the show *envar* command. The SRM environment variables are specific to the SRM console.

Table B-2   Environment Variable Summary

Environment Variable	Function
auto_action	Specifies the console's action at power-up, a failure, or a reset.
bootdef_dev	Specifies the default boot device string.
boot_osflags	Specifies the default operating system boot flags.
com2_baud	Changes the default baud rate of the COM2 serial port.
console	Specifies the device on which power-up output is displayed (serial terminal or graphics monitor).
cpu_enabled	Enables or disables a specific secondary CPU.
ew*0_mode	Specifies the connection type of the default Ethernet controller.
ew*0_protocols	Specifies network protocols for booting over the Ethernet controller.
kbd_hardware_ type	Specifies the default console keyboard type.
kzpsa*_host_id	Specifies the default value for the KZPSA host SCSI bus node ID.
language	Specifies the console keyboard layout.

Continued on next page

Table B-2   Environment Variable Summary (Continued)

Environment Variable	Function
memory_test	Specifies the extent to which memory will be tested. For DIGITAL UNIX systems only.
ocp_text	Overrides the default OCP display text with specified text.
os_type	Specifies the operating system and sets the appropriate console interface.
pci_parity	Disables or enables parity checking on the PCI bus.
pk*0_fast	Enables fast SCSI mode.
pk*0_host_id	Specifies the default value for a controller host bus node ID.
pk*0_soft_term	Enables or disables SCSI terminators on systems that use the QLogic ISP1020 SCSI controller.
sys_model_num	Displays the system model number and computes certain information passed to the operating system. Must be restored after a PCI motherboard is replaced.
sys_serial_num	Restores the system serial number. Must be set if the system motherboard is replaced.
sys_type	Displays the system type and computes certain information passed to the operating system. Must be restored after a PCI motherboard is replaced.
tga_sync_green	Specifies the location of the SYNC signal generated by the DIGITAL ZLXp-E PCI graphics accelerator option.
tt_allow_login	Enables or disables login to the SRM console firmware on other console ports.

B.3 Recording Environment Variables

You can make copies of the table below to record environment variable settings for specific systems. Write the system name in the column provided. Enter the show* command to list the system settings.

Table B-3   Environment Variables Worksheet

Environment Variable	System Name	System Name	System Name
auto_action			
bootdef_dev			
boot_osflags			
com2_baud			
console			
cpu_enabled			
ew*0_mode			
ew*0_protocols			
kbd_hardware_type			
kzpsa*_host_id			
language			
memory_test			
ocp_text			
os_type			
pci_parity			
pk*0_fast			
pk*0_host_id			
pk*0_soft_term			

Table B-3   Environment Variables Worksheet (Continued)

Environment Variable	System Name	System Name	System Name
pk*0_soft_term			
sys_model_num			
sys_serial_num			
sys_type			
tga_sync_green			
tt_allow_login			

Appendix C

Operating the System Remotely

This appendix describes how to use the remote console monitor (RCM) to monitor and control the system remotely .

C.1 RCM Console Overview

The remote console monitor (RCM) is used to monitor and control the system remotely. The RCM resides on the server control module and allows the system administrator to connect remotely to a managed system through a modem, using a serial terminal or terminal emulator.

The RCM has special console firmware that is used to remotely control an AlphaServer system. The RCM firmware resides on an independent microprocessor. It is not part of the SRM console that resides in the flash ROM. The RCM firmware has its own command interface that allows the user to perform the tasks that can usually be done from the system's serial console terminal. RCM console commands are used to reset, halt, and power the system on or off, regardless of the operating system or hardware state. The RCM console commands are also used to monitor the power supplies, temperature, and fans.

The user can enter the RCM console either remotely or through the local serial console terminal. Once in command mode, the user can enter commands to control and monitor the system.

- To enter the RCM console remotely, the user dials in through a modem, enters a password, and then types a special escape sequence that invokes RCM command mode. You must set up the modem before you can dial in remotely. See Section C.1.1.

- To enter the RCM console locally, the user types the escape sequence at the SRM console prompt on the local serial console terminal.

The RCM also provides an autonomous dial-out capability when it detects a power failure within the system. When triggered, the RCM dials a paging service at 30-minute intervals until the administrator clears the alert within the RCM.

C.1.1 Modem Usage

To use the RCM to monitor a system remotely, first make the connections to the server control module, as shown below. Then configure the modem port for dial-in.

Figure C-1 RCM Connections

PK-0651-96
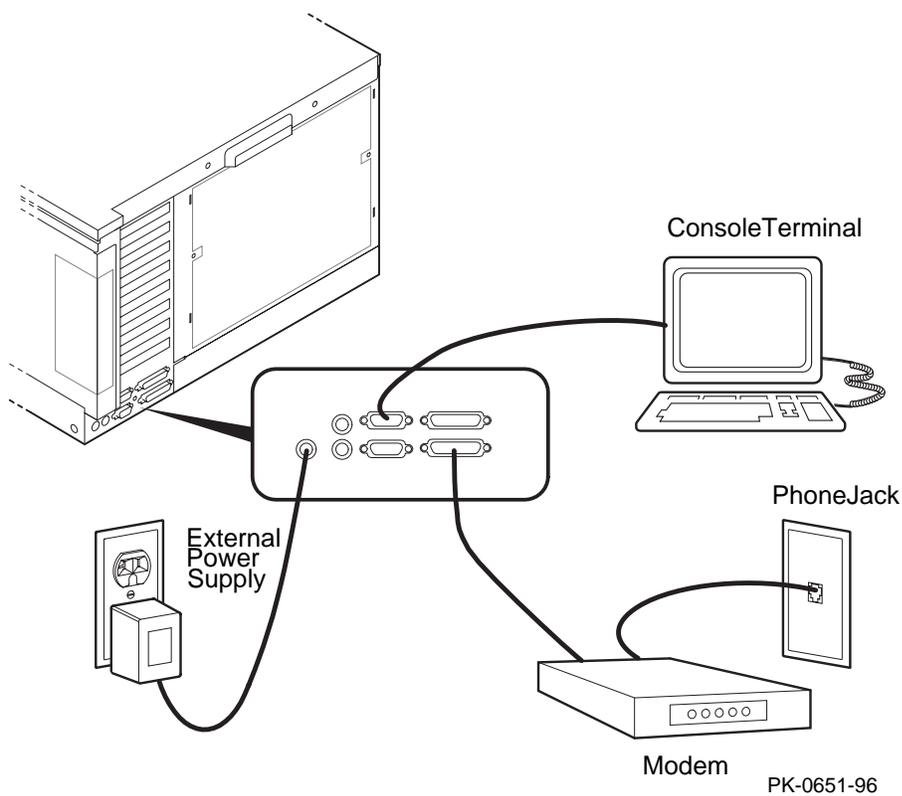
Modem Selection

The RCM requires a Hayes-compatible modem. The controls that the RCM sends to the modem have been selected to be acceptable to a wide selection of modems. The modems that have been tested and qualified include:

Motorola LifeStyle Series 28.8
AT&T DATAPORT 14.4/FAX
Zoom Model 360

The U.S. Robotics Sportster DATA/FAX MODEM is also supported, but requires some modification of the modem initialization and answer strings. See Section C.1.7.

Modem Configuration Procedure

1. Connect a Hayes-compatible modem to the RCM as shown in Figure C-1, and power up the modem.

2. From the local serial console terminal, enter the RCM firmware console by typing the following escape sequence:

   ```
   ^]^]rcm
   ```

   The character "^" is created by simultaneously holding down the Ctrl key and pressing the] key (right square bracket). The firmware prompt, RCM>, should now be displayed.

3. Enter a modem password with the **setpass** command. See Section C.1.3.14.

4. Enable the modem port with the **enable** command. See Section C.1.3.5.

5. Enter the **quit** command to leave the RCM console.

6. You are now ready to dial in remotely.

Dialing In to the RCM Modem Port

1. Dial the modem connected to the server control module. The RCM answers the call and after a few seconds prompts for a password with a "#" character.

2. Enter the password that was loaded using the **setpass** command. The user has three tries to correctly enter the password. On the third unsuccessful attempt, the connection is terminated, and as a security precaution, the modem is not answered again for 5 minutes.

   On successful entry of the password, the RCM banner message "RCM V1.0" is displayed, and the user is connected to the system COM1 port. At this point the local terminal keyboard is disabled except for entering the RCM console firmware. The local terminal displays all the terminal traffic going out to the modem.

3. To connect to the RCM firmware console, type the RCM escape sequence.

Refer to Example C-1 for an example of the modem dial-in procedure.

Example C-1   Sample Remote Dial-In Dialog

```
ATQ0V1E1S0=0    When modem dial-in connection is made, a screen display
OK              similar to this appears.
ATDT30167
CONNECT 9600
```

`#`	Enter password at this prompt.
`RCM V1.0`	RCM banner is displayed.
`^]^]rcm`	Enter the escape sequence after the banner is displayed. The escape sequence is not echoed on the terminal.
`RCM>`	RCM prompt is displayed. Commands to control and monitor the system can be entered.

Terminating a Modem Session

Terminate the modem session by executing a **hangup** command from the RCM console firmware. This will cleanly terminate the modem connection.

If the modem connection is terminated without using the **hangup** command, or if the line is dropped due to phone line problems, the RCM will detect carrier loss and initiate an internal **hangup** command. This process can take a minute or more, and the local terminal will be locked out until the auto hangup process completes.

If the modem link is idle for more than 20 minutes, the RCM initiates an auto hangup.

C.1.2 Entering and Leaving Command Mode

Use the default escape sequence to enter RCM command mode for the first time. You can enter RCM command mode from the SRM console level, the operating system level, or an application. The RCM quit command reconnects the terminal to the system console port.

Example C-2   Entering and Leaving RCM Command Mode

```
^]^]rcm              ❶
RCM>

RCM> quit            ❷
Focus returned to COM port
```

Entering the RCM Firmware Console

To enter the RCM firmware console, enter the RCM escape sequence. See ❶ in Example C-2 for the default sequence.

The escape sequence is not echoed on the terminal or sent to the system. Once in the RCM firmware console, the user is in RCM command mode and can enter RCM console commands.

Leaving Command Mode

To leave RCM command mode and reconnect to the system console port, enter the **quit** command, then press Return to get a prompt from the operating system or system console. (See ❷).

C.1.3 RCM Commands

The RCM commands summarized below are used to control and monitor a
system remotely.

Table C-1   RCM Command Summary

Command	Function
alert_clr	Clears alert flag, stopping dial-out alert cycle
alert_dis	Disables the dial-out alert function
alert_ena	Enables the dial-out alert function
disable	Disables remote access to the modem port
enable	Enables remote access to the modem port
hangup	Terminates the modem connection
halt	Halts server
help or **?**	Displays the list of commands
poweroff	Turns off power to server
poweron	Turns on power to server
quit	Exits console mode and returns to system console port
reset	Resets the server
setesc	Changes the escape sequence for entering command mode
setpass	Changes the modem access password
status	Displays server's status and sensors

Command Conventions

- The commands are not case sensitive.
- A command must be entered in full.
- If a command is entered that is not valid, the command fails with the message:

    *** ERROR - unknown command ***

    Enter a valid command.

The RCM commands are described on the following pages.

C.1.3.1 alert_clr

The **alert_clr** command clears an alert condition within the RCM. The alert enable condition remains active, and the RCM will again enter the alert condition when it detects a system power failure.

```
RCM>alert_clr
```

C.1.3.2 alert_dis

The **alert_dis** command disables RCM dial-out capability. It also clears any outstanding alerts. The alert disable state is nonvolatile. Dial-out capability remains disabled until the **alert_enable** command is issued.

```
RCM>alert_dis
```

C.1.3.3 alert_ena

The **alert_ena** command enables the RCM to automatically dial out when it detects a power failure within the system. The RCM repeats the dial-out alert at 30-minute intervals until the alert is cleared. The alert enable state is nonvolatile. Dial-out capability remains enabled until the **alert_disable** command is issued.

```
RCM>alert_ena
```

In order for the **alert_enable** command to work, two conditions must be met:

- A modem dial-out string must be entered with the system console.
- Remote access to the RCM modem port must be enabled with the **enable** command.

If the **alert_enable** command is entered when remote access is disabled, the following message is returned:

```
*** error ***
```

C.1.3.4 disable

The **disable** command disables remote access to the RCM modem port.

```
RCM>disable
```

The module's remote access default state is DISABLED. The modem enable state is nonvolatile. When the modem is disabled, it remains disabled until the **enable** command is issued. If a modem connection is in progress, entering the **disable** command terminates it.

C.1.3.5 enable

The **enable** command enables remote access to the RCM modem port. It can take up to 10 seconds for the **enable** command to be executed.

```
RCM>enable
```

The module's remote access default state is DISABLED.

The modem enable state is nonvolatile. When the modem is enabled, it remains enabled until the **disable** command is issued.

The **enable** command can fail for two reasons:

- There is no modem access password configured.

- The modem is not connected or is not working properly.

If the **enable** command fails, the following message is displayed:

```
*** ERROR - enable failed ***
```

C.1.3.6 hangup

The **hangup** command terminates the modem session. When this command is issued, the remote user is disconnected from the server. This command can be issued from either the local or remote console.

```
RCM>hangup
```

C.1.3.7 halt

The **halt** command attempts to halt the managed system. It is functionally
equivalent to pressing the Halt button on the system operator control panel to the
"in" position and then releasing it to the "out" position. The RCM console firmware
exits command mode and reconnects the user's terminal to the server's COM1 serial
port.

```
RCM>halt
Focus returned to COM port
```

NOTE: Pressing the Halt button has no effect on systems running Windows NT.

C.1.3.8 help or ?

The **help** or **?** command displays the RCM firmware command set.

C.1.3.9 poweroff

The **poweroff** command requests the RCM module to power off the system. It is
functionally equivalent to turning off the system power from the operator control
panel.

```
RCM>poweroff
```

If the system is already powered off, this command has no effect.

The external power to the RCM must be connected in order to power off the system
from the RCM firmware console. If the external power supply is not connected, the
command will not power the system down, and displays the message:

```
*** ERROR ***
```

C.1.3.10 poweron

The **poweron** command requests the RCM module to power on the system. For the system power to come on, the following conditions must be met:

- AC power must be present at the power supply inputs.
- The DC On/Off button must be in the "on" position.
- All system interlocks must be set correctly.

The RCM firmware console exits command mode and reconnects the user's terminal to the system console port.

```
RCM>poweron
Focus returned to COM port
```

*NOTE: If the system is powered off with the DC On/Off button, the system will not power up. The RCM will not override the "off" state of the DC On/Off button. If the system is already powered on, the **poweron** command has no effect.*

C.1.3.11 quit

The **quit** command exits the user from command mode and reconnects the user's terminal to the system console port. The following message is displayed:

```
Focus returned to COM port
```

The next display depends on what the system was doing when the RCM was invoked. For example, if the RCM was invoked from the SRM console prompt, the console prompt will be displayed when you enter a carriage return. Or, if the RCM was invoked from the operating system prompt, the operating system prompt will be displayed when you enter a carriage return.

C.1.3.12 reset

The **reset** command requests the RCM module to perform a hardware reset. It is functionally equivalent to pressing the Reset button on the system operator control panel.

```
RCM>reset
Focus returned to COM port
```

The following events occur when the **reset** command is executed:

- The system restarts and the system console firmware reinitializes.

- The console exits RCM command mode and reconnects the user's terminal to the server's COM1 serial port.

- The power-up messages are displayed, and then the console prompt is displayed or the operating system boot messages are displayed, depending on the state of the Halt button.

C.1.3.13 setesc

The **setesc** command allows the user to reset the default escape sequence for entering console mode. The escape sequence can be any character string. A typical sequence consists of 2 or more characters, to a maximum of 15 characters. The escape sequence is stored in the module's on-board NVRAM.

NOTE: If you change the escape sequence, be sure to record the new sequence. Although the module factory defaults can be restored if the user has forgotten the escape sequence, this involves accessing the server control module and moving a jumper.

The following sample escape sequence consists of five iterations of the Ctrl key and the letter "o".

```
RCM>setesc
^o^o^o^o^o
RCM>
```

If the escape sequence entered exceeds 15 characters, the command fails with the message:

```
*** ERROR ***
```

When changing the default escape sequence, avoid using special characters that are used by the system's terminal emulator or applications.

Control characters are not echoed when entering the escape sequence. To verify the complete escape sequence, use the **status** command.

C.1.3.14 setpass

The **setpass** command allows the user to change the modem access password that is prompted for at the beginning of a modem session. The password is stored in the module's on-board NVRAM.

```
RCM>setpass
new pass>*********
RCM>
```

The maximum password length is 15 characters. If the password entered exceeds 15 characters, the command fails with the message:

```
***  ERROR ***
```

The minimum password length is one character, followed by a carriage return. If only a carriage return is entered, the command fails with the message:

```
*** ERROR - illegal password ***
```

If the user has forgotten the password, a new password can be entered.

C.1.3.15 status

The **status** command displays the current state of the server's sensors, as well as the current escape sequence and alarm information.

```
RCM>status

Firmware Rev: V1.0
Escape Sequence: ^]^]RCM
Remote Access: ENABLE/DISABLE
Alerts: ENABLE/DISABLE
Alert Pending: YES/NO (C)
Temp (C): 26.0
RCM Power Control: ON/OFF
External Power: ON
Server Power: OFF

RCM>
```

The status fields are explained in Table C-2.

Table C-2 RCM Status Command Fields

Item	Description
Firmware Rev:	Revision of RCM firmware.
Escape Sequence:	Current escape sequence to enter RCM firmware console.
Remote Access:	Modem remote access state. (ENABLE/DISABLE)
Alerts:	Alert dial-out state. (ENABLE/DISABLE)
Alert Pending:	Alert condition triggered. (YES/NO)
Temp (C):	Current system temperature in degrees Celsius.
RCM Power Control:	Current state of RCM system power control. (ON/OFF)
External Power:	Current state of power from external power supply to server control module. (ON/OFF)
Server Power:	Current state of system power. (ON/OFF)

C.1.4 Dial-Out Alerts

The RCM can be configured to automatically dial out through the modem (usually to a paging service) when it detects a power failure within the system. When a dial-out alert is triggered, the RCM initializes the modem for dial-out, sends the dial-out string, hangs up the modem, and reconfigures the modem for dial-in. The RCM and modem must continue to be powered, and the phone line must remain active, for the dial-out alert function to operate.

Example C-3 Configuring the Modem for Dial-Out Alerts

```
P00>>> set rcm_dialout "ATDTstring#;"   ❶

RCM>enable
RCM>status
.
.
Remote Access:   ENABLE                  ❷
.
RCM>alert_ena                            ❸
```

Example C-4 Typical RCM Dial-Out Command

```
P00>>> set rcm_dialout "ATXDT9,15085553333,,,,,,5085553332#;"
```

Use the **show** command to verify the RCM dial-out string:

```
P00>>> show rcm_dialout
rcm_dialout        ATXDT9,15085553333,,,,,,5085553332#;
```

Enabling the Dial-Out Alert Function:

1. Enter the **set rcm_dialout** command, followed by a dial-out alert string, from the SRM console (see ❶ in Example C-3).

    The *string* is a modem dial-out character string, not to exceed 47 characters, that is used by the RCM when dialing out through the modem. See the next topic for details on composing the modem dial-out string.

2. Enter the RCM firmware console and enter the **enable** command to enable remote access dial-in. The RCM firmware **status** command should display "Remote Access: ENABLE." (See ❷.)

3. Enter the RCM firmware **alert_ena** command to enable outgoing alerts. (See ❸.)

Composing a Modem Dial-Out String

The modem dial-out string emulates a user dialing an automatic paging service. Typically, the user dials the pager phone number, waits for a tone, and then enters a series of numbers.

The RCM dial-out string (Example C-4) has the following requirements:

- The entire string following the **set rcm_dialout** command must be enclosed by quotation marks.

- The characters ATDT must be entered after the opening quotation marks. Do not mix case. Enter the characters either in all uppercase or all lowercase.

- Enter the character X if the line to be used also carries voice mail. Refer to the example that follows.

- The valid characters for the dial-out string are the characters on a phone keypad: 0–9, *, and #. In addition, a comma (,) requests that the modem pause for 2 seconds, and a semicolon (;) is required to terminate the string.

Elements of the Dial-Out String

ATXDT	AT = Attention X = Forces the modem to dial "blindly" (not look for a dial tone). Enter this character if the dial-out line modifies its dial tone when used for services such as voice mail. D = Dial T = Tone (for touch-tone) , = Pause for 2 seconds.
9,	In the example, "9" gets an outside line. Enter the number for an outside line if your system requires it.
15085553333	Dial the paging service.
,,,,,,	Pause for 12 seconds for paging service to answer.
5085553332#	"Message," usually a call-back number for the paging service.
;	Return to console command mode. Must be entered at end of string.

C.1.5 Resetting the RCM to Factory Defaults

If the escape sequence has been forgotten, you can reset the controller to factory settings.

Reset Procedure

1. Power down the AlphaServer system and access the server control module, as follows:

    Expose the PCI bus card cage. Remove three Phillips head screws holding the cover in place and slide it off the drawer. If necessary, remove several PCI and EISA options from the bottom of the PCI card cage until you have enough space to access the server control module.

2. Unplug the external power supply to the server control module.

    Locate the password and option reset jumper. The jumper number, which is etched on the board, depends on the revision of the server control module.

    NOTE: If the RCM section of the server control module does not have an orange relay, the jumper number is J6. If the RCM section of the server control module has an orange relay, the jumper number is J7.

3. Move the jumper so that it is sitting on both pins.

4. Replace any panels or covers as necessary so you can power up the system. Press the Halt button and then power up the system to the SRM console prompt. Powering up with the password and option reset jumper in place resets the escape sequence, password, and modem enable states to the factory default.

5. When the console prompt is displayed, power down the system and move the password and option reset jumper back onto the single pin.

6. Replace any PCI or EISA modules you removed and replace the PCI bus card cage cover.

7. Power up the system to the SRM console prompt and type the default escape sequence to enter RCM command mode:

    `^]^]RCM`

8. Configure the module as desired. You must reset the password and modem enable states in order to enable remote access.

C.1.6 Troubleshooting Guide

Table C-3 lists a number of possible causes and suggested solutions for symptoms you might see.

Table C-3   RCM Troubleshooting

Symptom	Possible Cause	Suggested Solution
The local terminal will not communicate with the system or the RCM.	System and terminal baud rate set incorrectly.	Set the system and terminal baud rates to 9600 baud.
	Cables not correctly installed.	Review external cable installation.
RCM will not answer when the modem is called.	Modem cables may be incorrectly installed.	Check modem phone lines and connections.
	RCM remote access is disabled.	Enable remote access.
	RCM does not have a valid password set.	Set password and enable remote access.
	The local terminal is currently in the RCM console firmware.	Issue a **quit** command on the local terminal.
	On power-up, the RCM defers initializing the modem for 30 seconds to allow the modem to complete its internal diagnostics and initialization.	Wait 30 seconds after powering up the system and RCM before attempting to dial in.
	Modem may have had power cycled since last being initialized or modem is not set up correctly.	Enter **enable** command from RCM console.

Table C-3 RCM Troubleshooting (Continued)

Symptom	Possible Cause	Suggested Solution
After the system and RCM are powered up, the COM port seems to hang and then starts working after a few seconds.	This delay is normal behavior.	Wait a few seconds for the COM port to start working.
RCM installation is complete, but system will not power up.	RCM Power Control: is set to DISABLE.	Enter RCM console and issue the **poweron** command.
New password, escape sequence, and modem enable state are forgotten when system and RCM module are powered down.	The password and option reset jumper is still installed. If the RCM section of the server control module does not have an orange relay, the jumper number is J6. If it does have an orange relay, the number is J7.	After resetting RCM to factory defaults, move the jumper so that it is sitting on only one pin.
The remote user sees a "+++" string on the screen.	The modem is confirming whether the modem has really lost carrier. This occurs when the modem sees an idle time, followed by a "3," followed by a carriage return, with no subsequent traffic. If the modem is still connected, it will remain so.	This is normal behavior.
The message "unknown command" is displayed when the user enters a carriage return by itself.	The terminal or terminal emulator is including a linefeed character with the carriage return.	Change the terminal or terminal emulator setting so that "new line" is not selected.

Continued on next page

Table C-3 RCM Troubleshooting (Continued)

Symptom	Possible Cause	Suggested Solution
Cannot enable modem or modem will not answer.	The modem is not configured correctly to work with the RCM.	Modify the modem initialization and/or answer string.

C.1.7 Modem Dialog Details

This section provides further details on the dialog between the RCM and the modem and is intended to help you reprogram your modem if necessary.

Phases of Modem Operation

The RCM is programmed to expect specific responses from the modem during four phases of operation:

- Initialization

- Ring detection

- Answer

- Hang-up

The initialization and answer command strings are stored in the RCM NVRAM. The factory default strings are:

Initialization string: `AT&F0EVS0=0S12=50<cr>`

Answer string   `ATXA<cr>`

NOTE: All modem commands must be terminated with a <cr> character (0x0d hex).

Initialization

The RCM initializes the modem to the following configuration:

Factory defaults (`&F0`)

No Echo (`E`)

Numeric response codes (`V`)

No Auto Answer (`S0=0`)

Guard-band = 1 second (`S12=50`)

Fixed modem-to-RCM baud rate

Connect at highest possible reliability and speed

The RCM expects to receive a "`0<cr>`" (OK) in response to the initialization string. If it does not, the **enable** command will fail.

This default initialization string works on a wide variety of modems. If your modem does not configure itself to these parameters, the initialization string will need to be modified. See the topic in this section entitled Modifying Initialization and Answer Strings.

Ring Detection

The RCM expects to be informed of an in-bound call by the modem signaling the RCM with the string, "2<cr>" (RING).

Answer

When the RCM receives the ring message from the modem, it responds with the answer string. The "X" command modifier used in the default answer string forces the modem to report simple connect, rather than connect at *xxxx*. The RCM expects a simple connect message, "1<cr>" (CONNECTED). If the modem responds with anything else, the RCM forces a hang-up and initializes the modem.

The default answer string is formatted to request the modem to provide only basic status. If your modem does not provide the basic response, the answer string, and/or initialization string will need to be modified. See the topic in this section entitled Modifying Initialization and Answer Strings.

After receiving the "connect" status, the modem waits for 6 seconds and then prompts the user for a password.

Hangup

When the RCM is requested to hang up the modem, it forces the modem into command mode and issues the **hangup** command to the modem. This is done by pausing for a minimum of the guard time, sending the modem "+++". When the modem responds with "0<cr>" (OK), the hang-up command string is sent. The modem should respond with "3<cr>" (NO CARRIER). After this interchange, the modem is reinitialized in preparation for the next dial-in session.

RCM/Modem Interchange Overview

Table C-4 summarizes the actions between the RCM and the modem from initialization to hangup.

Table C-4   RCM/Modem Interchange Summary

Action	Data to Modem	Data from Modem
Initialization command	AT&F0EVS0=0S12=50<cr>	
Initialization successful		0<cr>
Phone line ringing		2<cr>
RCM answering	ATXA<cr>	
Modem successfully connected		1<cr>
Force modem into command mode	<guard_band>+++	
Modem in command mode		0<cr>
Hangup	ATH<cr>	
Successful hangup		3<cr>

Modifying Initialization and Answer Strings

The initialization and answer strings are stored in the RCM's NVRAM. They come pre-programmed to support a wide selection of modems. In the circumstance where the default initialization and answer strings do not set the modem into the desired mode, the following SRM **set** and **show** commands are provided to enable the user to define and examine the initialization and answer strings.

To replace the initialization string:

```
P00>>> set rcm_init "new_init_string"
```

To replace the answer string:

```
P00>>> set rcm_answer "new_answer_string"
```

To display all the RCM user settable strings:

```
P00>>> show rcm*
rcm_answer  ATXA
rcm_dialout
rcm_init    AT&F0EVS0=0S12=50
P00>>>
```

Initialization and Answer String Substitutions

The RCM default initialization and answer strings are as follows:

Initialization String: "AT&F0EVS0=0S12=50"

Answer String: "ATXA"

The following modem requires a modified answer string.

	Initialization String	Answer String
USRobotics Sportster 28,800 Data/Fax Modem	RCM default	"ATX0&B1&A0A"

Index