AlphaServer DS15 and AlphaStation DS15

Service Guide

Order Number: EK-DS150-SG. A01

This manual is intended for service providers and self-maintenance customers for DS15 systems.

Hewlett-Packard Company

October 2003

FCC Notice

This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules, which are designed to provide reasonable protection against such radio frequency interference.

Operation of this equipment in a residential area may cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

Any modifications to this device—unless expressly approved by the manufacturer—can void the user’s authority to operate this equipment under part 15 of the FCC rules.

Modifications

The FCC requires the user to be notified that any changes or modifications made to this device that are not expressly approved by Hewlett-Packard Company may void the user's authority to operate the equipment.

Cables

Connections to this device must be made with shielded cables with metallic RFI/EMI connector hoods in order to maintain compliance with FCC Rules and Regulations.

Taiwanese Notice

警告使用者:

這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。

Japanese Notice

この装置は，情報処理等電波障害自主規制協議会（VCCI）基準に基づくクラス A 情報装置です。この装置を家庭環境で使用すると電波障害を引き起こすことがあります。この場合には，使用者が適切な対策を講じるよう要求されることがあります。

Canadian Notice

This Class A digital apparatus meets all requirements of the Canadian Interference-Causing Equipment Regulations.

Avis Canadien

Cet appareil numérique de la classe A respecte toutes les exigences du Règlement sur le matériel brouilleur du Canada.

European Union Notice

Products with the CE Marking comply with both the EMC Directive (89/336/EEC) and the Low Voltage Directive (73/23/EEC) issued by the Commission of the European Community.

Compliance with these directives implies conformity to the following European Norms (in brackets are the equivalent international standards):

EN55022 (CISPR 22) - Electromagnetic Interference

EN50082-1 (IEC801-2, IEC801-3, IEC801-4) - Electromagnetic Immunity

EN60950 (IEC950) - Product Safety

Warning!

This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Achtung!

Dieses ist ein Gerät der Funkstörgrenzwertklasse A. In Wohnbereichen können bei Betrieb dieses Gerätes Rundfunkstörungen auftreten, in welchen Fällen der Benutzer für entsprechende Gegenmaßnahmen verantwortlich ist.

Attention!

Ceci est un produit de Classe A. Dans un environnement domestique, ce produit risque de créer des interférences radioélectriques, il appartiendra alors à l'utilisateur de prendre les mesures spécifiques appropriées.

Contents

Chapter 3 Power-Up Diagnostics and Display

Chapter 4 SRM Console Diagnostics

Chapter 5 Error Logs

Chapter 6 System Configuration and Setup

Chapter 7 Using the Remote Management Console

Chapter 8 FRU Removal and Replacement

Appendix A Jumpers on System Motherboard

Appendix B Isolating Failing DIMMs

Examples

Figures

Tables

Index

Preface

Intended Audience

This manual is for service providers and self-maintenance customers for *AlphaServer* DS15 systems.

Document Structure

This manual uses a structured documentation design. Topics are organized into small sections, usually consisting of two facing pages. Most topics begin with an abstract that provides an overview of the section, followed by an illustration or example. The facing page contains descriptions, procedures, and syntax definitions.

This manual contains eight chapters and two appendixes.

- **Chapter 1, System Overview**, provides an overview of the system.
- **Chapter 2, Troubleshooting**, describes the starting points for diagnosing problems on AlphaServer DS15 systems and also provides information resources.
- **Chapter 3, Power-Up Diagnostics and Display**, explains the power-up process and RMC, SROM, and SRM power-up diagnostics.
- **Chapter 4, SRM Console Diagnostics**, describes troubleshooting with the SRM console.
- **Chapter 5, Error Logs**, explains how to interpret error logs reported by the operating system.
- **Chapter 6, System Configuration and Setup**, describes how to configure and set up a DS15 system.
- **Chapter 7, Using the Remote Management Console**, explains how to manage the system through the remote management console (RMC).
- **Chapter 8, FRU Removal and Replacement**, describes the procedures for removing and replacing Field Replaceable Units (FRUs) on *AlphaServer* DS15 systems.
- **Appendix A, Jumpers on System Motherboard**, provides detailed information on the configuration of jumpers on the system motherboard

- **Appendix B, Isolating Failing DIMMs**, explains how to manually isolate a failing DIMM from the failing address and failing data bits.

Documentation Titles

hp AlphaServer DS15 and AlphaStation DS15 Documentation

Title	Order Number
User Documentation Kit	QA-72XAA-G8
DS15 AlphaServer and DS15 AlphaStation Owner's Guide	EK–DS150–OG
AlphaServer DS15 Quick Setup	EK–DS150–IG
AlphaServer DS15 Floor Stand Kit	EK–DS150–FS
DS15 AlphaServer and DS15 AlphaStation Service Guide	EK–DS150–SG
CD-ROM Installation Guide	EK–DS152–CD
AlphaServer DS15 Release Notes	EK–DS150–RN

Information on the Internet

Visit the *AlphaServer* Web site at http://h18002.www1.hp.com/alphaserver/ for service tools and more information about the *AlphaServer* DS15 and *AlphaStation* DS15 system.

Chapter 1
System Overview

This chapter provides an overview of the system including:

- System Enclosure Configurations
- Common Components
- Front View
- Top View
- Rear Ports and Slots
- Network Connection
- Operator Control Panel
- System Motherboard
- PCI Slots
- Storage Cage Options
- Console Terminal
- Power Connection
- System Access Lock

1.1 System Enclosure Configurations

The DS15 family consists of a rackmounted system, a standalone pedestal system, and a desktop system. All have similar features, components, capabilities and options; the desktop system will be shown throughout this manual in illustrations and examples.

Figure 1–1 DS15 Rackmounted and Pedestal System

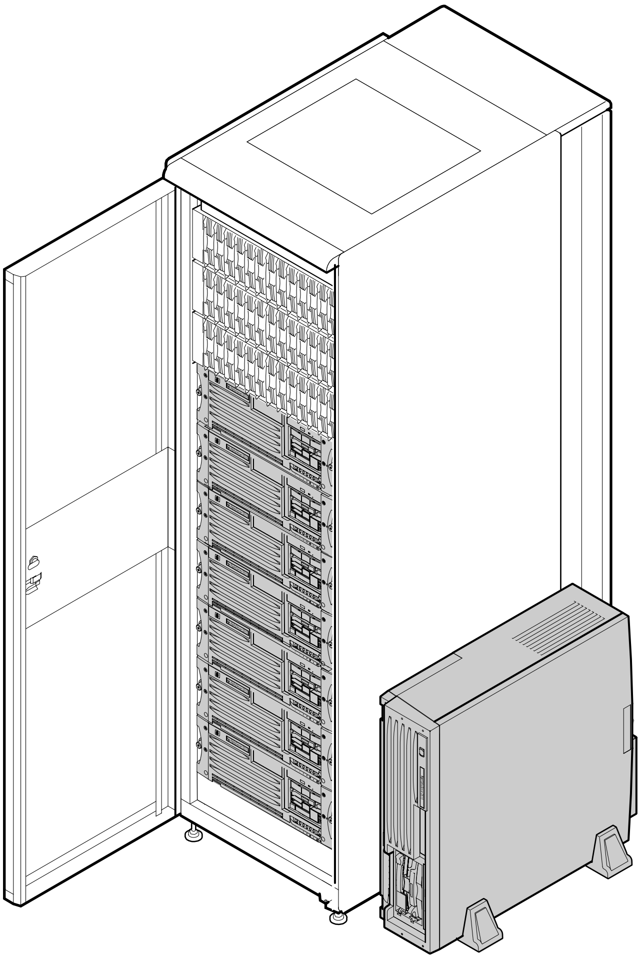


Figure 1–2 DS15 Desktop System with Internal Storage Cage Option

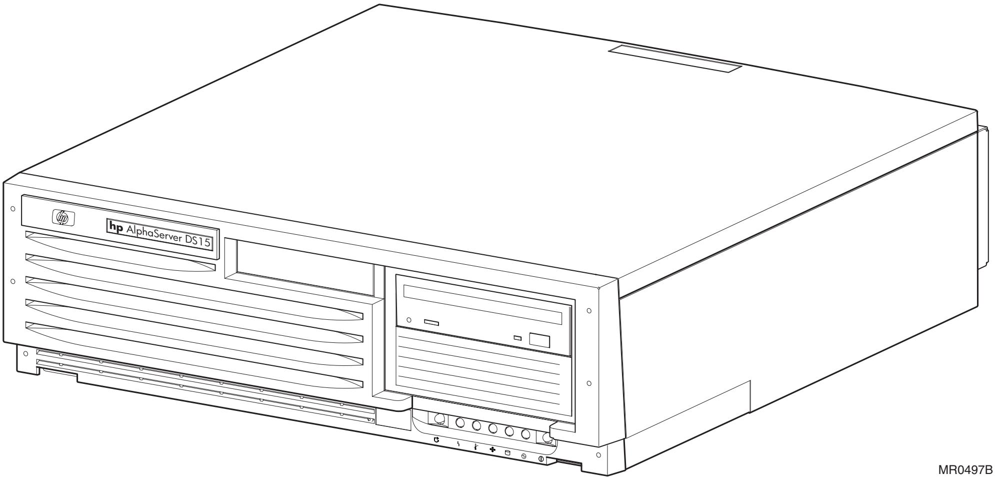
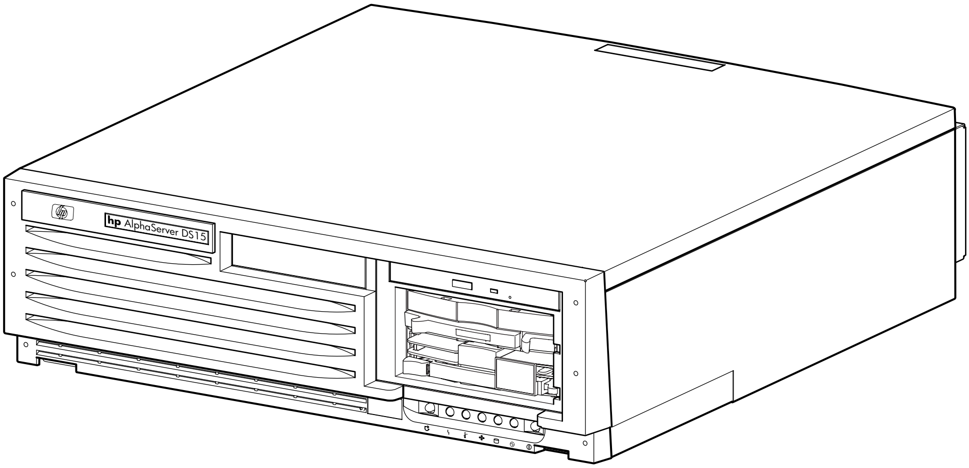


Figure 1–3 DS15 Desktop System with Front Access Storage Cage Option



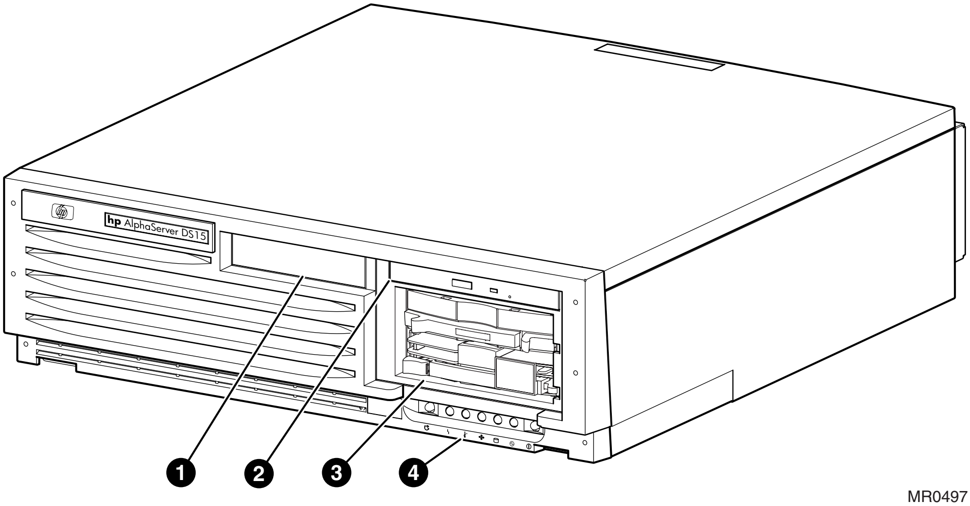
1.2 Common Components

The basic building block of *AlphaServer* DS15 systems is the system enclosure chassis that houses the following common components.

- Alpha 1 GHz CPU with 2 MB onboard ECC cache
- 512-MB, 1 GB, or 2 GB SDRAM memory – expandable to 4 GB maximum memory capacity
- Onboard dual 10/100 BaseT Ethernet ports
- Four 64-bit PCI expansion slots
- Onboard Dual Channel Ultra160 SCSI controller
- Choice of storage cage subsystems:
    a. Internal Storage Cage with a maximum SCSI storage capacity of 218.4 GB
    b. Front Access Storage Cage with a maximum SCSI storage capacity of 510.4 GB
- Two serial ports:
    a. COM1 port with RMC port with modem control and a full-duplex asynchronous communications port
    b. COM2 port with full-duplex asynchronous communications port
- PS/2-style keyboard port and mouse port
- 400W (120/240V, 60/50 Hz) power supply

1.3 Front View

Figure 1–4 Front View with Optional Front Access Storage Cage



❶ Center internal storage bay

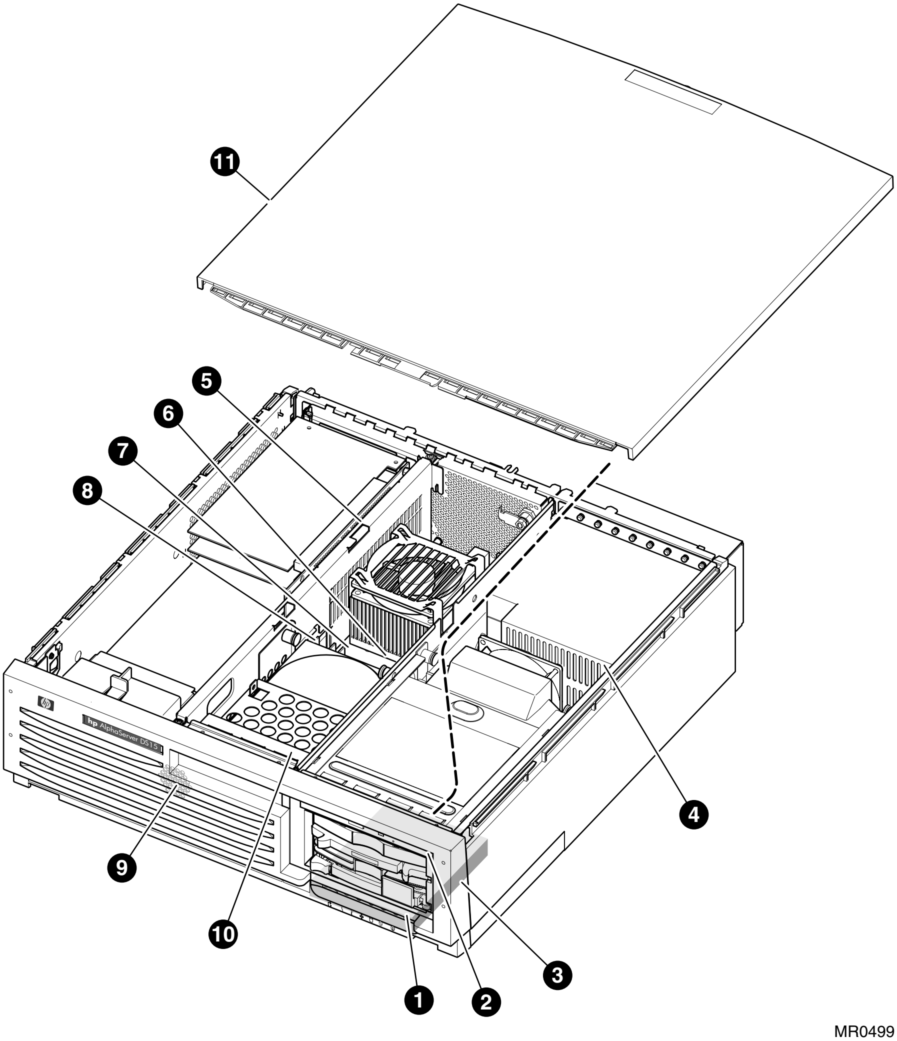
❷ DVD/CD-RW drive

❸ Disk storage

❹ Operator control panel

1.4 Top View

Figure 1–5 Top View



❶ Operator Control Panel

❷ DVD/CD-RW drive

❸ Internal disk drive

❹ Power supply

❺ PCI riser

❻ CPU

❼ System motherboard

❽ Memory

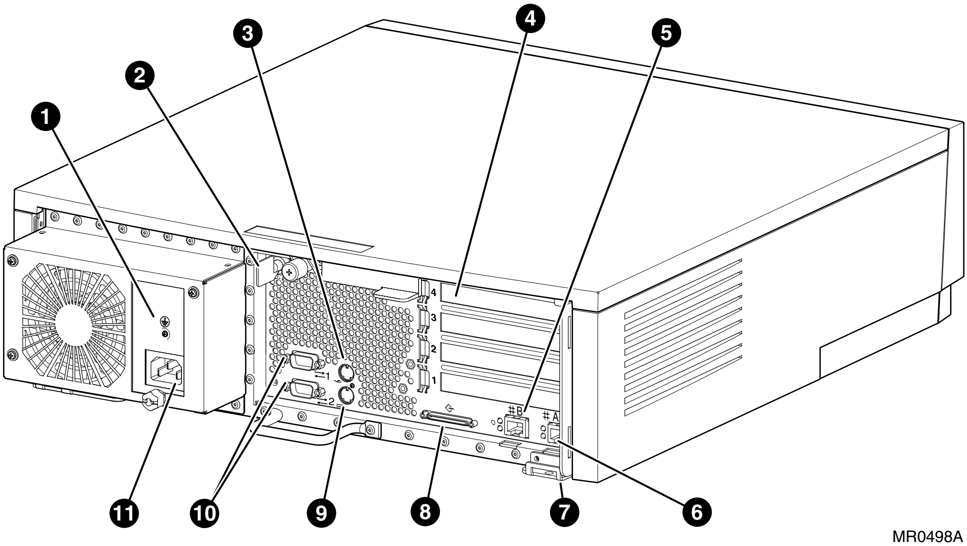
❾ Speaker (hidden)

❿ Center internal storage bay

⓫ Cover

1.5 Rear Ports and Slots

Figure 1–6 Rear Ports and Slots



❶ Power supply ground

❷ Key

❸ Mouse connector

❹ PCI Slots

❺ Ethernet port B

❻ Ethernet port A

❼ Cable run hook

❽ SCSI connector

❾ Keyboard connector

❿ COM 1 serial port (top), COM 2 serial port (bottom)

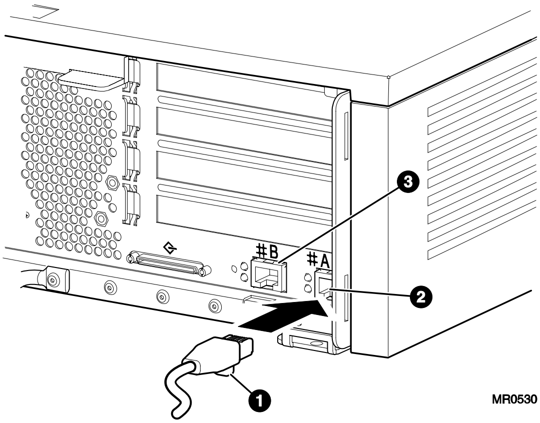
⓫ Power connector

1.6 Network Connections

There are two onboard Ethernet network connectors on the rear of the DS15 system.

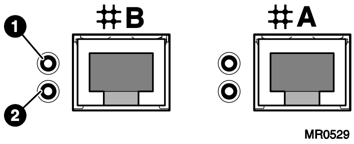
The DS15 system has dual onboard 10/100 BaseT Ethernet ports. You can connect to either or both.

Figure 1–7 Ethernet Network Connection



Connect the Ethernet cable ❶ to either Ethernet connector A ❷ or B ❸.

Figure 1–8 Network LED indicators



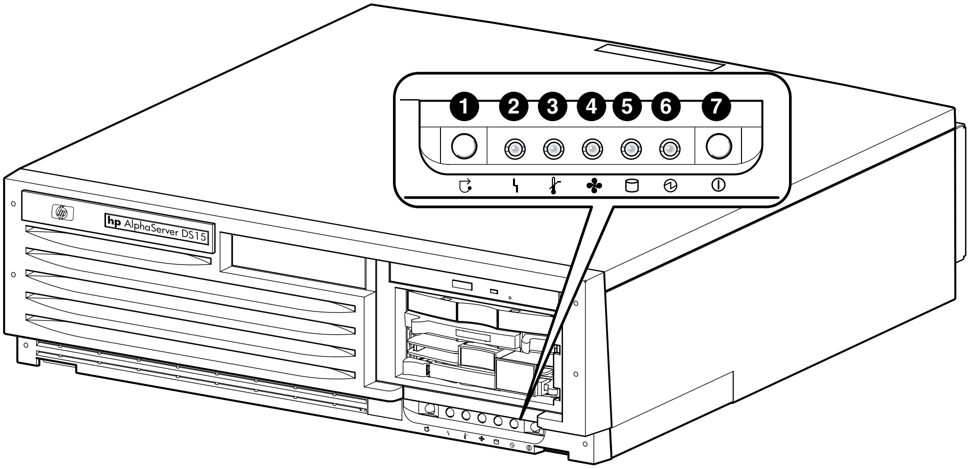
The LEDs to the left of each Ethernet connector indicate its status.

❶ LED Speed/Activity; indicates activity through the connection.
❷ LED Link indicator; network connection exists when this is lit.

1.7 Operator Control Panel

The control panel provides system controls and status indicators. The controls are the Power and Halt/Reset buttons. The panel has a green power LED, a green disk activity indicator LED, and three amber diagnostic LEDs.

Figure 1–9 Operator Control Panel



❶ Halt/Reset button

❷ Amber system fault LED

❸ Amber over temperature fault LED

❹ Amber fan fault LED

❺ Green disk activity LED

❻ Green system power LED

❼ System Power Switch (On/Off)

NOTE: *Jumper J22 (pins 13 – 14) must be installed for the halt/reset button to function as a reset button.*

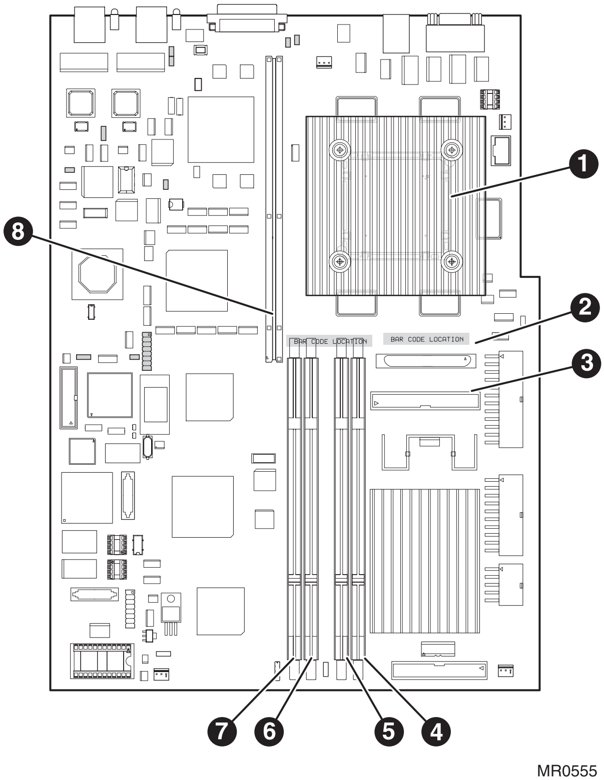
1.7.1 Remote Commands

Commands issued from the remote management console (RMC) can be used to reset, halt, and power the system on or off. For information on RMC, see Chapter 7.

RMC Command	Function
Power on	Turns on power. Emulates pressing the Power button to the On position.
Power off	Turns off power. Emulates pressing the Power button to the Off position.
Halt	Halts the system.
Halt in	Halts the system and causes the halt to remain asserted.
Halt out	Releases a halt created with **halt in**.
Reset	Resets the system.

1.8 System Motherboard

Figure 1–10 System Motherboard



❶ CPU

❷ Internal SCSI connector

❸ IDE connector

❹ Memory DIMM slot - array 2, DIMM 2

❺ Memory DIMM slot - array 0, DIMM 0

❻ Memory DIMM slot - array 2, DIMM 3

❼ Memory DIMM slot - array 0, DIMM 1

❽ Slot for PCI riser

1.8.1 CPU

The CPU microprocessor is a superscalar pipelined processor packaged in a 675-pin LGA carrier. The CPU has the ability to issue up to four instructions during each CPU clock cycle and a peak instruction execution rate of four times the CPU clock frequency.

1.8.2 DIMMS

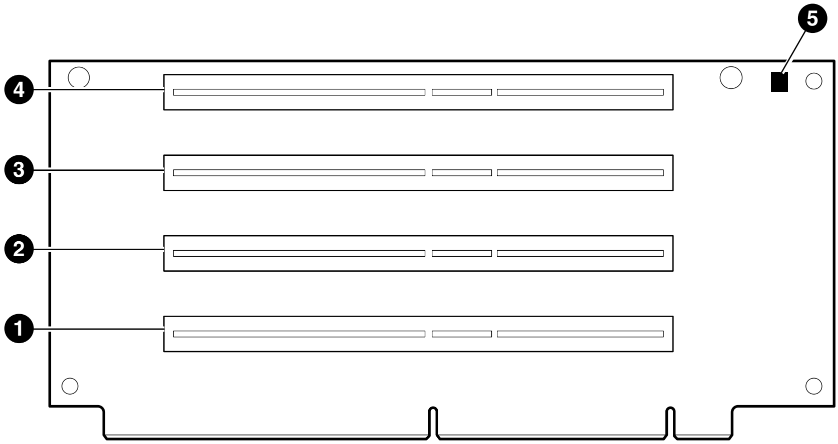
The *AlphaServer* DS15 system supports up to two pairs of 200-pin synchronous DIMMs. Supported DIMM sizes are 256 MB, 512 MB, and 1 GB, allowing memory to be configured from 512 MB to 4096 MB.

1.8.3 PCI

The *AlphaServer* DS15 system supports two PCI busses, one for the onboard integrated I/O and the other controls the four expansion slots through the PCI riser card.

1.9 Slots on the PCI Riser Card

Figure 1–11 Slots on the PCI Riser Card



❶ Slot 1 – 66/33 MHz, 3.3v

❷ Slot 2 – 66/33 MHz, 3.3v

❸ Slot 3 – 33 MHz, 3.3v

❹ Slot 4 – 33 MHz, 3.3v

❺ LED – connected to +5 VAUX

Table 1–1 How Physical I/O Slots Map to Logical Slots

Port	Hose	Physical Slot	SRM Logical Slot
A	2	1	7
		2	8
		3	9
		4	10

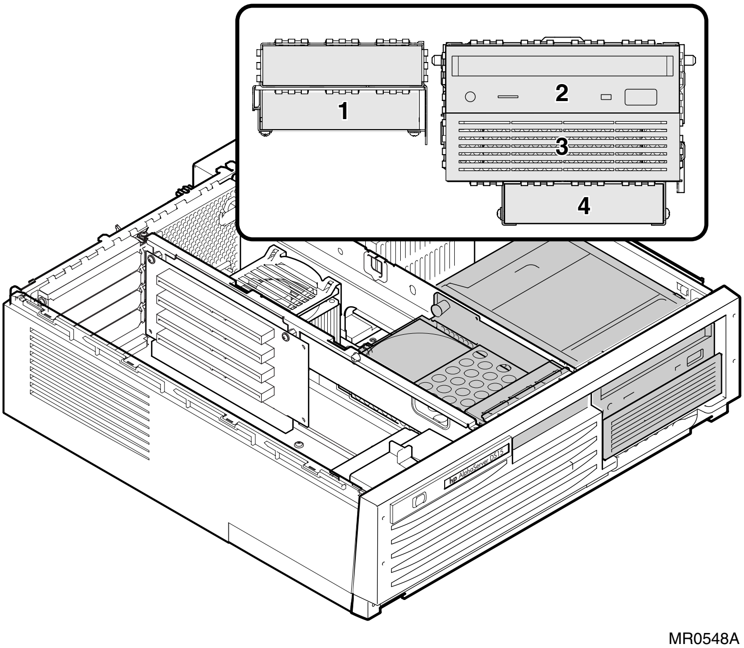
1.10 Storage Cage Options

The *AlphaServer* DS15 system comes with either an internal storage cage or a front access storage cage.

1.10.1 Internal Storage Cage

Systems configured with an internal storage cage includes a half-height DVD/CD-RW drive and a half-height bay for a disk, DVD/CD-RW, or tape drive. The cage supports three 3.5-inch x 1-inch hard disk drives *or* two internal 3.5 inch x 1-inch hard disk drives and one 5.25-inch x 1.6-inch removable media device.

Figure 1–12 Internal Storage Cage Configuration



❶ Center internal storage bay

❷ DVD/CD-RW drive

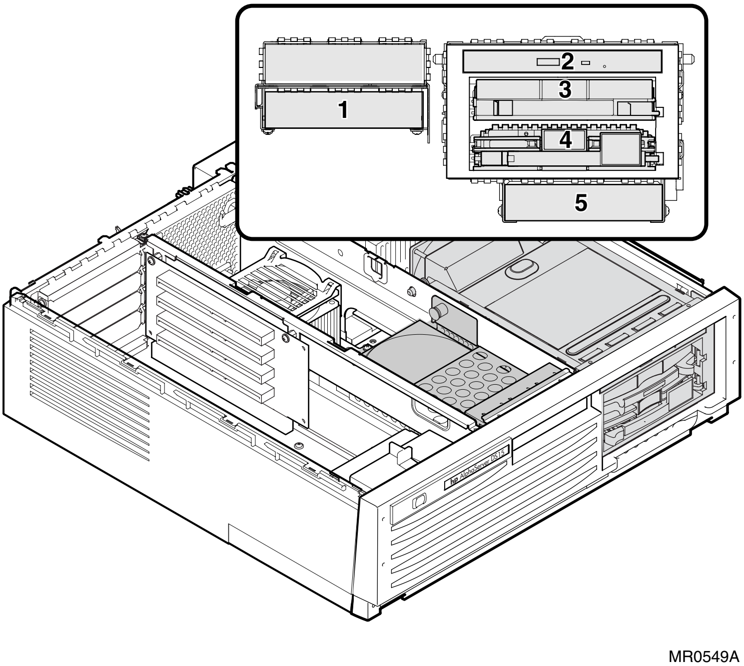
❸ DVD/CD-RW or internal drive bay (disk or tape)

❹ Internal drive bay

1.10.2 Front Access Storage Cage

Systems configured with a front access storage cage includes a slim-line DVD/CD-RW drive and two 3.5-inch x 1-inch hard disk drive bays or one front access universal tape drive bay. The cage supports two front access 3.5-inch x 1-inch hard disk drives and two internal 3.5-inch x 1-inch hard disk drives *or* one front access universal tape drive (AIT or DAT) and two internal disk drives.

Figure 1–13 Front Access Storage Cage Configuration



❶ Center internal storage bay

❷ DVD/CD-RW drive

❸ Universal drive bay

❹ Universal drive bay

❺ Internal drive bay

1.11 Console Terminal

The console terminal can be a serial (character cell) terminal connected to the COM1 port or a VGA monitor connected to a VGA adapter.

Figure 1–14 Console Terminal Connected to COM Port

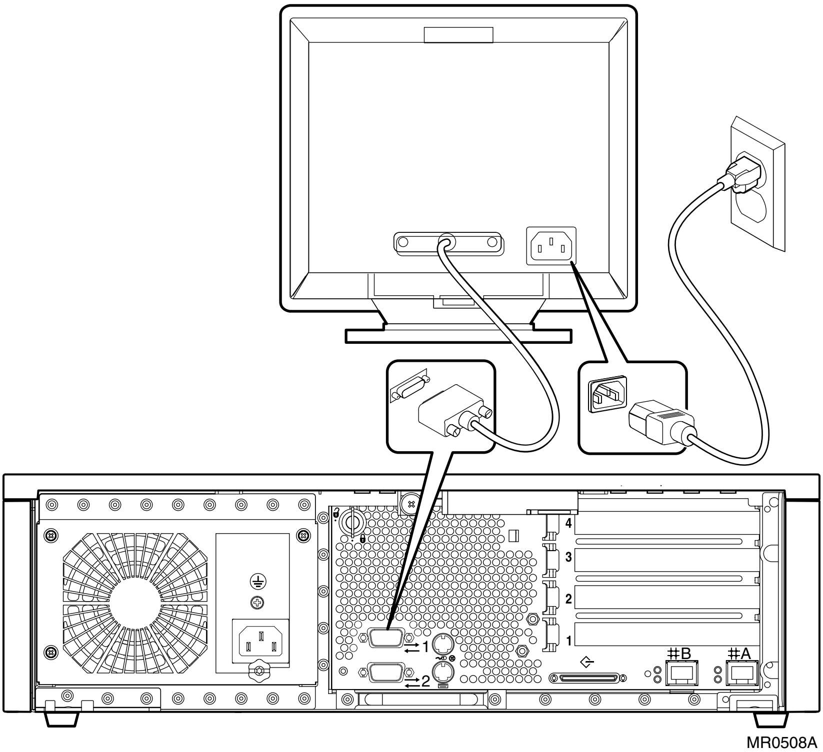
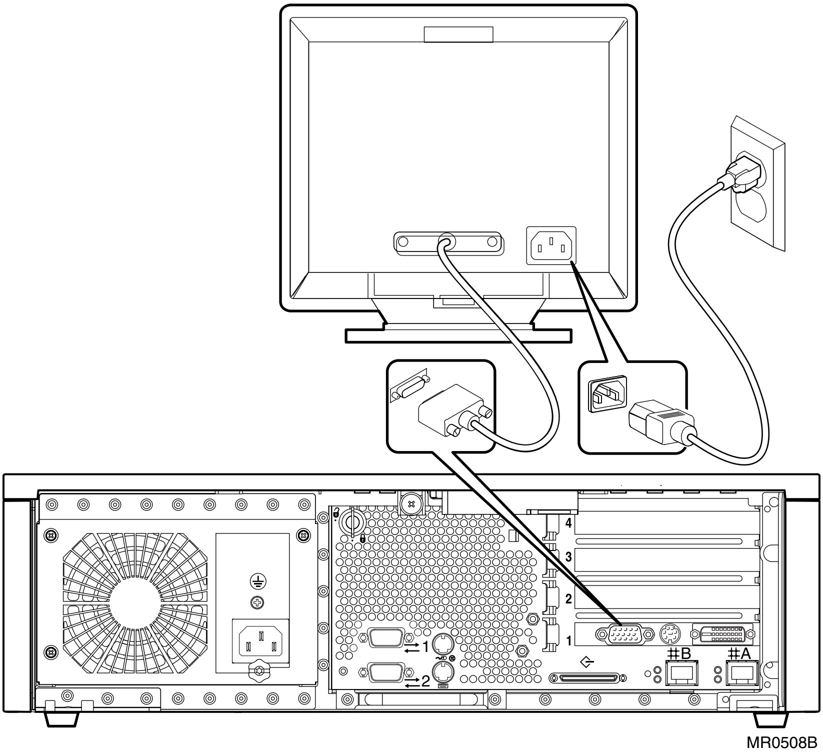


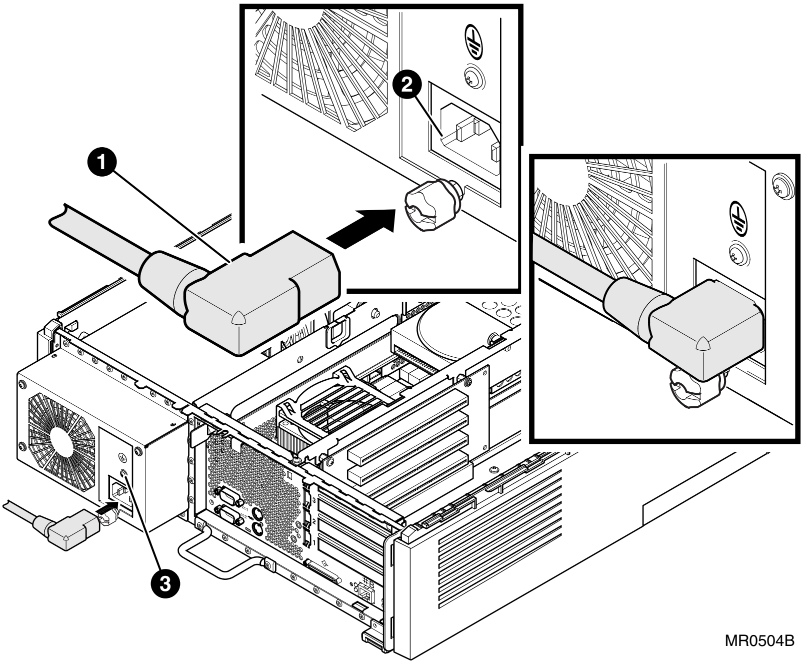
Figure 1–15 Console Terminal Connected to Optional Video Card



1.12 Power Connection

Figure 1–16 shows the power connection for a desktop system.

Figure 1–16 Connecting the Power for the Desktop

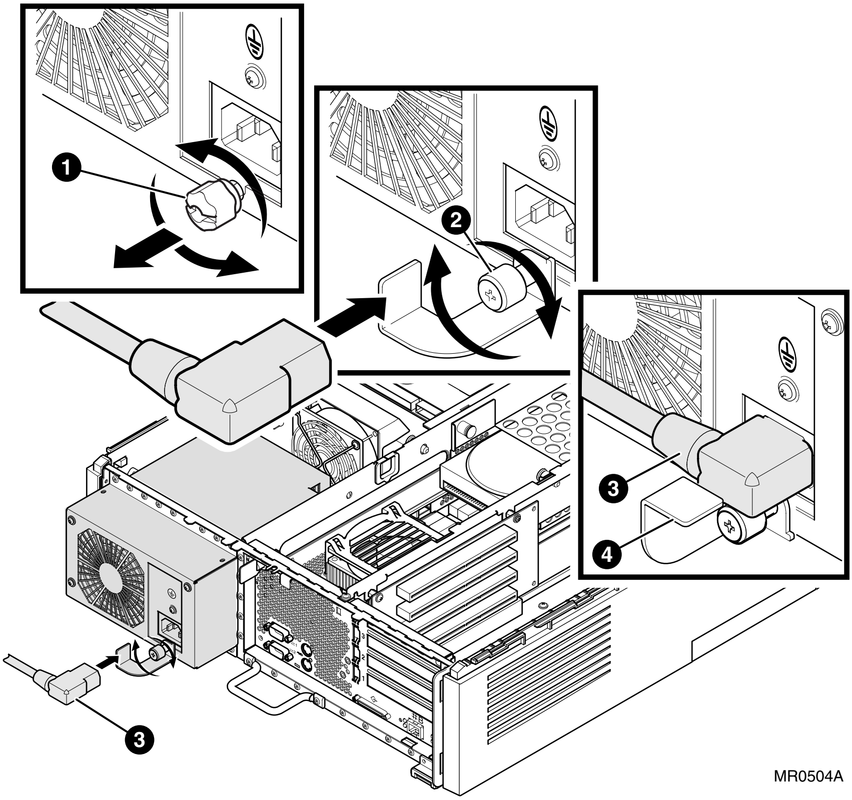


❶ Power cord

❷ Power receptacle

❸ Ground screw

Figure 1–17 Connecting the Power for a Rackmount System



❶ Thumbscrew

❷ Power cord bracket with attached screw

❸ Power cord

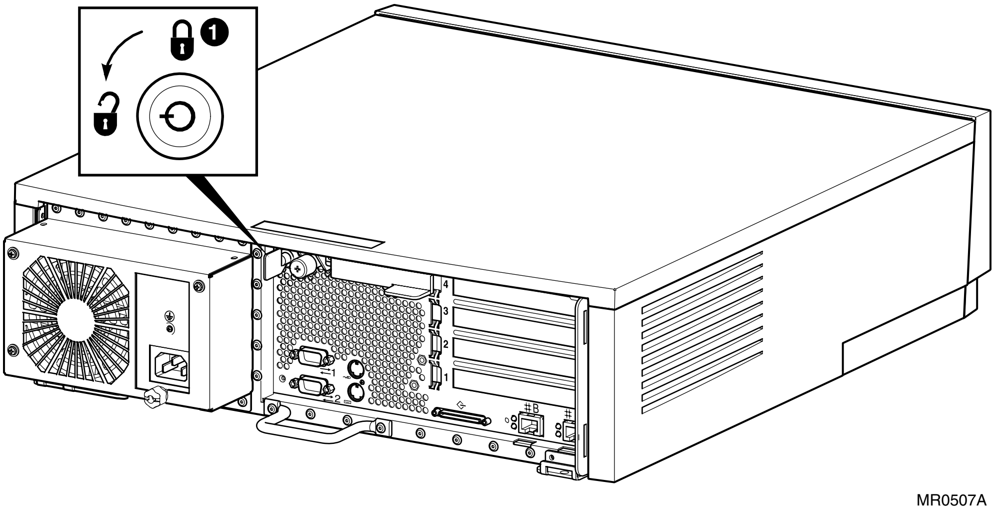
❹ Power cord bracket

To connect the power cord, loosen the thumbscrew, plug the cord in, rotate the bracket so that it supports the power cord plug, and tighten the attached screw.

1.13 System Access Lock

The system enclosure has a key lock ➊ for security, as shown in the figure. If you wish to limit access to the inside of the enclosure, keep the system locked and the key in a secure location.

Figure 1–18 System Access Lock



Chapter 2
Troubleshooting

This chapter describes the starting points for diagnosing problems on *AlphaServer* DS15 systems. The chapter also provides information resources.

- Questions to Consider
- Diagnostic Categories
- Fail-Safe Booter Utility
- Updating Firmware
- Service Tools and Utilities
- Q-Vet Installation Verification
- Information Resources

2.1 Questions to Consider

Before troubleshooting any system problem, first check the site maintenance log for the system's service history.

Be sure to ask the system manager the following questions:

- Has the system been used and did it work correctly?
- Have changes to hardware or updates to firmware or software been made to the system recently? If so, are the revision numbers compatible for the system? (Refer to the system release notes.)
- What is the current state of the system?
  - If the operating system is down, but you are able to access the SRM console, use the console environment diagnostic tools, including the Operator Control Panel (OCP) LEDs and SRM commands.
  - If you are unable to access the SRM console, enter the Remote Management Console (RMC) command-line interface (CLI) and issue commands to determine the hardware status. See Chapter 7.
  - If the operating system has crashed and rebooted, the Computer Crash Analysis Tool (CCAT), the System Event Analyzer service tools (to interpret error logs), the SRM **crash** command, and operating system exercisers can be used to diagnose system problems.

2.2 Diagnostic Categories

System problems can be classified into the following categories. Using these categories, you can quickly determine a starting point for diagnosis and eliminate the unlikely sources of the problem.

The next several subsections group problems into one of several categories.

- Error beep codes
- Diagnostic LEDs on the OCP
- Power problems
- Problems getting to the console mode
- Problems reported by the console mode
- Boot problems
- Errors reported by the operating system
- Memory problems
- PCI bus problems
- SCSI problems
- Thermal problems and environmental status

WARNING: To prevent injury, access is limited to persons who have appropriate technical training and experience. Such persons are expected to understand the hazards of working within this equipment and take measures to minimize danger to themselves or others. These measures include:

1. **Remove any jewelry that may conduct electricity.**
2. **If accessing the system card cage, power down the system and wait 2 minutes to allow components to cool.**
3. **Wear an anti-static wrist strap when handling internal components.**

2.2.1 Error Beep Codes

Audible beep codes announce specific errors that might be encountered while the system is powering up. Table 2–1 identifies the error beep codes.

Table 2–1 Error Beep Codes

Beeps	Message/Meaning	Action to Repair
1	Done with execution; jumping to console	No action necessary.
1-3-3	No usable memory available	Check memory and memory configuration.
2-1-2	Configuration error detected	Check system configurations.
1-1-4	ROM checksum error detected	Replace the system board.
1-2-4	Bcache error detected	Possible CPU problem.

2.2.2 Diagnostic LEDs on the OCP

Diagnostic LEDs on the operator control panel indicate error conditions and power-up information.

Figure 2–1 LED Patterns during Power-Up

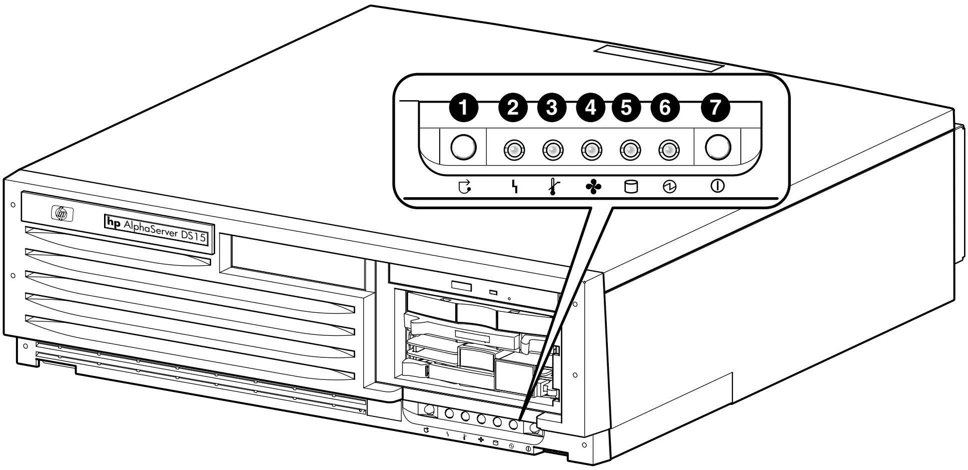


Table 2–2 OCP Switches

Switch	Function
❶	Halt/Reset
❼	System Power Switch (On/Off)

Table 2–3 OCP LED Indications

LED	Color	LED On Function
❻	Green	System power is on.
❺	Green	There is disk activity.
❹	Amber	There is a fan fault.
❸	Amber	The system is over temperature.
❷	Amber	There is a system fault.
❷ and ❹	Amber/blink in unison	RMC image is corrupted but the system is not in emergency runtime image recovery mode or emergency runtime image recovery mode has timed out. If recovery has timed out, unplug the system power cord and wait until the LED on the PCI Riser card turns off. Plug in the power cord and try again.
❷ ❸ ❹	Amber/blink in unison	System is in emergency runtime image recovery mode and is awaiting firmware update.
❷ ❸ ❹	Amber	RMC has failed or the system is configured for emergency runtime image recovery but is not powered on.
❷ ❸ ❹	Amber/blink sequentially	Firmware update is in progress.

2.2.3 Power Problems

Power problems can prevent the system from operating. Use the following table to troubleshoot these problems.

Table 2–4 Power Problems

If the power indicator is:	Check:
Off	Front-panel power switch Power at the wall receptacle AC cord Power cable connectors Unplug the power cord for 15 seconds, then reconnect.
On for a few seconds and then goes Off	Enter the RMC. Use the **poe** command to check for poweron errors, and use the **log** or **log #** command to check the event log for symptoms of failure. Make sure that all jumpers are in their default state.
On, but the monitor screen is black for approximately 40 seconds and then turns blue.	Monitor power indicator is On. Video cable is properly connected. SRM **console** environment variable setting. EV may not be set to graphics. **NOTE:** *A black raster is displayed if the **console** environment variable is set to serial mode rather than graphics mode.*
Off and system does not power on remotely via RMC	Front panel switch is in the On (depressed) position.

2.2.4 Problems Getting to Console Mode

Certain problems can prevent access to console mode. Use the following table to troubleshoot these problems.

Table 2–5 Problems Getting to Console Mode

Symptom	Action
Power-up screen is not displayed at system console.	Note any error beep codes and observe the OCP for a failure detected during self-tests. Check keyboard and monitor connections. Press the Return key. If the system enters console mode, check that the **console** environment variable is set correctly. If the console terminal is a VGA monitor, the **console** variable should be set to **graphics**. If it is a serial terminal, the console environment variable should be set to **serial**. If **console** is set to **serial**, the power-up screen is routed to the COM1 serial communication port and cannot be viewed from the VGA monitor. Try connecting a console terminal to the COM2 serial communication port. When using the COM2 port set the **console** environment variable to **serial**. Use RMC commands to determine status.

2.2.5 Problems Reported by the Console

The console may report certain problems. Use the following table to troubleshoot these problems.

Table 2–6 Problems Reported by the Console

Symptom	Action
Power-up tests do not complete.	Use error beep codes or console serial terminal to determine what error occurred.
	Check the power-up screen for error messages.
	Enter the RMC. Use the **poe** command to check for poweron errors, and use the **log** or **log #** command to check the event log for symptoms of failure.
Console program reports an error.	Interpret the error beep codes at power-up and check the power-up screen for a failure detected during self-tests.
	Examine the console event log (use the **more el** command) to check for error messages recorded during power-up.
	If the power-up screen or console event log indicates problems with mass storage devices or PCI devices, or if devices are missing from the **show config** or **show device** display, see Section 2.2.9 and 2.2.10.
	Enter the RMC and check the power-on errors “poe” and the event log “log, log #” for symptoms for failure.
	Use the SRM **test** command to verify the problem.

2.2.6 Boot Problems

Certain problems may interfere with the boot process. Use the following table to troubleshoot these problems.

Table 2–7 Boot Problems

Problem/Possible Cause	Action
Operating system (OS) software is not installed on the disk drive.	Install the operating system and license key.
Target boot device is not listed in the SRM **show device** or **show config** command.	Check the cables. Are the cables oriented properly and not cocked? Are there bent pins? Check all the SCSI devices for incorrect or conflicting IDs. Refer to the device's documentation.
	SCSI termination: The SCSI bus must be terminated at the end of the internal cable and at the last external SCSI peripheral.
	Review the position of all relevant jumpers.

System cannot find the boot device.	Use the SRM **show config** and **show device** commands. Use the displayed information to identify target devices for the **boot** command, and verify that the system sees all of the installed devices. If you are attempting to use bootp, first set the following variables as shown. (Replace ewa0 with the appropriate device designation.)
	`>>>set ewa0_inet_init BOOTP`
	`>>>set ewa0_protocols BOOTP`

System does not boot.	Verify that no unsupported adapters are installed.
Environment variables are incorrectly set.	This could happen if the main logic board has been replaced, which would cause a loss of the previous configuration information.
	Use the SRM **show** and **set** commands to check and set the values assigned to boot-related variables such as **auto_action**, **bootdef_dev**, and **boot_osflags.**
System will not boot over the network.	For problems booting over a network, check the **ew*0_protocols**, **ei*0_protocols** or **eg*0_protocols** environment variable settings: Systems booting from a *Tru64 UNIX* server should be set to **bootp**; systems booting from an

Problem/Possible Cause	Action
	OpenVMS server should be set to **mop**. Run the **test** command to check that the boot device is operating. Check ei*0_mode. Refer to Table 6-1, SRM Environment Variables.

2.2.7 Errors Reported by the Operating System

The operating system may hang, crash, or log errors. Use the following table to troubleshoot these problems.

Table 2–8 Errors Reported by the Operating System

Symptom	Action
System is hung or has crashed.	If possible, halt the system by using either the halt/reset button or the RMC **halt** command. (Jumper J22 pins 13-14 must be removed. If jumper J22 is installed, you will reset the system and loose system context so that no crash can be acquired.) Then enter the SRM **crash** command and examine the crash dump file. Refer to the *Guide to Kernel Debugging* (AA-PS2TD-TE) for information on using the *Tru64 UNIX* Crash utility.
Errors have been logged and the operating system is up.	Examine the operating system error log files. Contact HP Services.

2.2.8 Memory Problems

Memory problems may affect system performance. Use the following table to troubleshoot these problems.

Table 2–9 Memory Testing

Symptom	Action
DIMMs ignored by system, or system unstable. System hangs or crashes.	Ensure that each memory array has identical DIMMs installed.
DIMMs failing memory power-up self-test.	Replace the DIMMs that the SROM has isolated on power up. See Example 2–1.
DIMMs may not have ECC bits.	Some third-party DIMMs may not be compatible with DS15 systems. Ensure that the DIMMs are qualified.
Noticeable performance degradation. The system may appear hung or run very slowly.	This could be a result of hard single-bit ECC errors on a particular DIMM. Check the error logs for memory errors. Ensure that the memory DIMMs are qualified.

Example 2–1 Memory Sizing

```
Memory sizing in progress
Memory configuration in progress
Testing AAR2
Memory data test in progress
Memory data path error

ErrAdr: 00000000.00000000
Expect: 00000000.00000001
Actual: 00000000.00000000
XORval: 00000000.00000001

Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory initialization
Failed DIMM 3
Loading console
Code execution complete (transfer control)
```

2.2.9 PCI Bus Problems

PCI bus problems at startup are usually indicated by the inability of the system to detect the PCI device. Use the following steps to diagnose the likely cause of PCI bus problems.

1. Five volt PCI adapters are not allowed.
2. Confirm that the PCI option module is supported and has the correct firmware and software versions.
3. Confirm that the PCI option module and any cabling are properly seated.
4. Check for a bad PCI slot by moving the last installed PCI option module to a different slot.
5. Contact HP Service to replace the PCI riser card.

PCI Parity Error

Some PCI devices do not implement PCI parity, and some have a parity generating scheme that may not comply with the PCI specification. In such cases, the device should function properly if parity is not checked.

Parity checking can be turned off with the **set pci_parity off** command so that false PCI parity errors do not result in machine check errors. However, if you disable PCI parity, no parity checking is implemented for any device. Turning off PCI parity is therefore not recommended or supported.

2.2.10 SCSI Problems

SCSI problems are generally manifested as data corruption, boot problems, or poor performance.

Do the following:

- Check SCSI bus termination and relevant SCSI jumpers.
- Ensure that all disks have a unique ID.
- Invoke "run bios" and check or configure SCSI devices.
- Cable is properly seated at system board or option connector.
- Bus must be terminated at last device on cable or at physical cable end.
- No terminators in between.
- Old 50-pin (narrow) devices must be connected with wide-to-narrow adapter (SN-PBXKP-BA). Do not cable from the connector on the card.
- Using 50-pin devices on the bus may significantly degrade performance.
- Any external drives must be connected to the external port on the rear of the system or to their associated card. These cards must have no internal drives connected to them.
- Ultra-wide SCSI has strict bus length requirements.
- SCSI bus itself cannot handle internal plus external cable.
- Connection of internal SCSI drives in either the front access or internal storage cage **is not supported**.

2.2.11 Thermal Problems and Environmental Status

Overtemperature conditions can cause the system to shut down.

The DS15 system operates in an ambient temperature range of 10ºC–40ºC. An internal sensor monitors the system temperature and shuts down the system if maximum limits are exceeded. If the system shuts down unexpectedly:

- Ensure that the side cover (pedestal) or top cover (rack) are properly secured.
- Verify that the ambient temperature does not exceed the specified limits.
- Make sure there are no airflow obstructions at the front or rear of the system.
- Check to see that the cables inside the system are properly dressed. A dangling cable can impede airflow to the system.

Troubleshooting with the show power Command

The SRM console **show power** command can help you determine if environmental problems necessitate the replacement of a power supply, system fan or fans, or the motherboard.

Show power indicates:	Action
Bad voltage	Replace the power supply and or the system motherboard. Contact HP Services.
Bad fan	Replace the indicated fan or fans. Contact HP Services.
Bad temperature	The problem could be a bad fan or an obstruction to the airflow. Check the airflow first. If there is no obstruction, contact HP Services to replace the bad fan.

Troubleshooting with RMC Commands

The RMC **status** command displays the system status and the current RMC settings. See Section 7.6.1 for more information.

The RMC **env** command provides a snapshot of the system environment. See Section 7.6.2 for more information.

The **log** command prints out a brief summary of the last 10 system events that have been logged. Issuing the **log** command followed by a number (0-9) provides detailed information about the selected system event (0 = most recent event).

2.3 Fail-Safe Booter Utility

The fail-safe booter utility (FSB) is another variant of the SRM console. The FSB provides an emergency recovery mechanism if the firmware image contained in flash memory becomes corrupted. You can run the FSB and boot another image from a CD-ROM or network that is capable of reprogramming the flash ROM.

Use the FSB when one of the following failures at power-up prohibits you from getting to the console program:

- Firmware image in flash memory corrupted
- Power failure or accidental power-down during a firmware upgrade
- Error in the nonvolatile RAM (NVRAM) file
- Incorrect environment variable setting
- Driver error

2.3.1 Starting the FSB Automatically

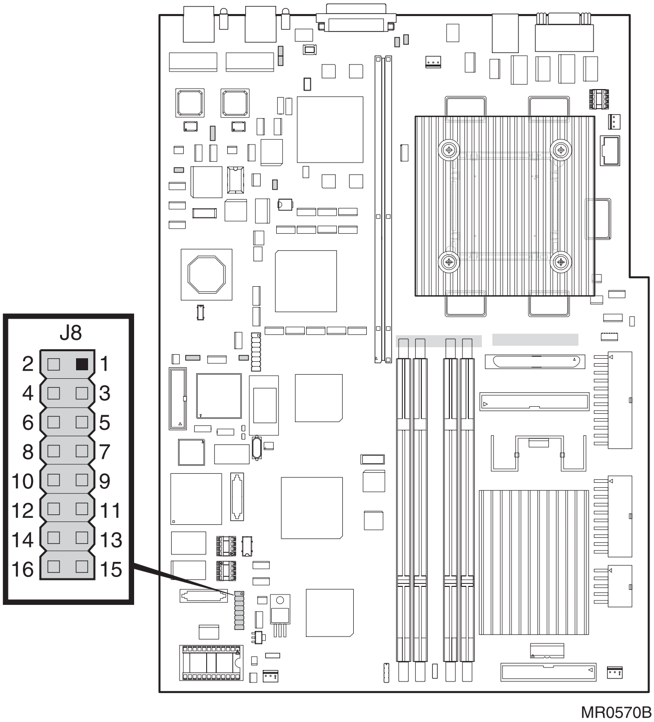
If the firmware image is unavailable when the system is powered on or reset, the FSB runs automatically.

1. Reset the system to restart the FSB. The FSB loads from the flash.
2. Update the firmware as described in Section 2.4.

2.3.2 Starting the FSB Manually

1. Power the system off, unplug the AC power cord, and remove the cover.
2. Insert jumper J8 over pins 1 – 2 on the system motherboard. See Figure 2–2 for a location.
3. Reconnect the AC power cord and reinstall the system cover.
4. Power up the system to the FSB console.

Figure 2–2 FSB Switch "On" Setting



2.3.3 Required Firmware

The required firmware for your system is preloaded onto the flash ROM. Copies of the firmware files are included on your distribution CD. You can also download the latest firmware files from the Alpha systems firmware Web site:

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

The utilities that are used to reload or update the firmware need to find the files on a CD.

2.4 Updating Firmware

Be sure to read the information on starting the FSB utility before continuing with this section.

Updating the Console Firmware

Perform the following steps to update the console firmware. Refer to Example 2–2.

1. Insert the Alpha Firmware CD into the DVD/CD-RW drive.
2. At the SRM console prompt, issue the **>>>b dqa0** command.
3. At the UPD> prompt, enter the **update SRM** command.

After the update has completed, enter the **exit** command to exit the utility.

Example 2–2 Running LFU

```
>>> boot dqa0

Checking dqa0.0.0.13.0 for the option firmware files. . .
dqa0.0.0.13.0 has no media present or is disabled via the RUN/STOP switch
Checking dqa1.1.0.13.0 for the option firmware files. . .
Checking dva0.0.0.1000.0 for the option firmware files. . .

Option firmware files were not found on CD or floppy.
If you want to load the options firmware,
please enter the device on which the files are located(ewa0),
or just hit <return> to proceed with a standard console update:

***** Loadable Firmware Update Utility *****
-----------------------------------------------------------------------------
Function    Description
-----------------------------------------------------------------------------
 Display     Displays the system's configuration table.
 Exit        Done exit LFU (reset).
 List        Lists the device, revision, firmware name, and update revision.
 Update      Replaces current firmware with loadable data image.
 Verify      Compares loadable and hardware images.
 ? or Help   Scrolls this function table.
-----------------------------------------------------------------------------

 UPD> update srm

Confirm update on:
SRM              [Y/(N)] y

WARNING: updates may take several minutes to complete for each device.

                          DO NOT ABORT!

SRM             Updating to X6.6-1977...  Verifying X6.6-1977...  PASSED.
```

```
 UPD> exit
 Do you want to do a manual update? [y/(n)] n

UPD> list

Device          Current Revision       Filename          Update Revision

FSB                T6.6-6              fsb_fw               T6.6-8
SRM                T6.6-7              srm_fw               T6.6-7
booter             V0.5-6              booter_fw         No Update Available

rt                 V0.6-3              rt_fw                V0.6-3
srom               V1.0-1              srom_fw              V1.0-1
tig                 1.9                tig_fw                1.9

UPD> u fsb

Confirm update on:
FSB              [Y/(N)]y

WARNING: updates may take several minutes to complete for each device.

                          DO NOT ABORT!

FSB              Updating to V6.6-8...  Verifying T6.6-8...  PASSED.

UPD> exit

 Initializing....
```

2.5 Service Tools and Utilities

This section lists some of the tools and utilities available for acceptance testing and diagnosis and gives recommendations for their use.

2.5.1 Error Handling/Logging Tools (System Event Analyzer)

The operating systems provide fault management error detection, handling, notification, and logging.

The primary tool for error handling is System Event Analyzer (SEA), a fault analysis utility designed to analyze both single and multiple error/fault events. SEA uses error/fault data sources other than the traditional binary error log. See Chapter 5 for more information.

2.5.2 Loopback Tests

Internal and external loopback tests are used to test the I/O components and adapter cards. The loopback tests are a subset of the SRM diagnostics.

Use loopback tests to isolate problems with the COM2 serial port, the parallel port, and Ethernet controllers. See the **test** command in Chapter 4 for instructions on performing loopback tests.

2.5.3 SRM Console Commands

SRM console commands are used to set and examine environment variables and device parameters. For example, the **show configuration** and **show device** commands are used to examine the configuration, and the **set *envar* and show *envar*** commands are used to set and view environment variables.

SRM commands are also used to invoke ROM-based diagnostics and to run native exercisers. For example, the **test** and **sys_exer** commands are used to test the system.

See Chapter 4 for information on running console exercisers. See Chapter 6 for information on configuration-related console commands and environment variables. See Chapter 7 for a list of console commands used most often on *AlphaServer* DS15 systems.

2.5.4 Remote Management Console (RMC)

The remote management console is used for managing the server either locally or remotely. It also plays a key role in error analysis by passing error log information to the dual-port RAM (DPR), which is shared between the RMC and the system motherboard logic, so that this information can be accessed by the system. RMC also controls the diagnostic LEDs on the Operator Control Panel (OCP). RMC has a command-line interface from which you can enter a few diagnostic commands.

RMC can be accessed as long as the power cord for a working supply is plugged into the AC wall outlet and a console terminal is attached to the system. This feature ensures that you can gather information when the operating system is down and the SRM console is not accessible. See Chapter 7.

2.5.5 Crash Dumps

For fatal errors, the operating systems save the contents of memory to a crash dump file. This file can be used to determine why the system crashed.

The Computer Crash Analysis Tool (CCAT) is the primary crash dump analysis tool for analyzing crash dumps on Alpha systems. CCAT compares the results of a crash dump with a set of rules. If the results match one or more rules, CCAT notifies the system user of the cause of the crash and provides information to avoid similar crashes in the future.

2.6 Q-Vet Installation Verification

CAUTION: Customers are not authorized to access, download, or use Q-Vet. Q-Vet is for use by HP engineers to verify the system installation. Misuse of Q-Vet may result in loss of customer data.

Q-Vet is the Qualification Verifier Exerciser Tool that is used by HP engineers to exercise systems under development. HP recommends running the latest Q-Vet released version to verify that hardware is installed correctly and is operational. Q-Vet does not verify specific operating system or layered product configurations.

The latest Q-Vet release, information, Release Notes, and documentation are located at http://cisweb.mro.cpqcorp.net/projects/qvet/.

HP recommends that Compaq Analyze be installed on the operating system prior to running Q-Vet.

CAUTION:

*Do **not** install the Digital System Verification Software (DECVET) on the system; use Q-Vet instead.*

Non-IVP Q-Vet scripts verify disk operation for some drives with "write enabled" techniques. These are intended for Engineering and Manufacturing Test. Run ONLY IVP scripts on systems that contain customer data or any other items that must not be written over. See the Q-Vet Disk Testing Policy Notice on the Q-Vet Web site for details. All Q-Vet IVP scripts use Read Only and/or File I/O to test hard drives. Floppy and tape drives are always write tested and should have scratch media installed

*Q-Vet **must** be de-installed upon completion of system verification.*

Swap or Pagefile Space

The system must have adequate swap space (on *Tru64 UNIX*) or pagefile space (on *OpenVMS*) for proper Q-Vet operation. You can set this up either before or after Q-Vet installation.

During initialization, Q-Vet will display a message indicating the minimum amount of swap/pagefile needed, if it determines that the system does not have enough. You can then reconfigure the system.

If you wish to address the swap/pagefile size before running Q-Vet, see the Swap/Pagefile Estimates on the Q-Vet Web site.

2.6.1 Installing Q-Vet

The procedures for installation of Q-Vet differ between operating systems. You must install Q-Vet on each partition in the system.

Install and run Q-Vet from the **SYSTEM** account on *OpenVMS* and the **root** account on *Tru64 UNIX*. Remember to install Q-Vet in each partition.

Tru64 UNIX

1. Make sure that there are no old Q-Vet or DECVET kits on the system by using the following command:
   `setld -i | grep VET`

   Note the names of any listed kits, such as OTKBASExxx etc., and remove the kits using **qvet_uninstall** if possible. Otherwise use the command
   `setld -d kit1_name kit2_name kit3_name`
2. Copy the kit tar file (*QVET_Vxxx.tar*) to your system.
3. Be sure that there is no directory named output. If so move to another directory or remove the output directory.
   `rm -r output`
4. Untar the kit with the command
   `tar xvf QVET_Vxxx.tar`
   Note: The case of the file name may be different depending upon how it was stored on the system. Also, you may need to enclose the file name in quotation marks if a semi-colon is used.
5. Install the kit with the command
   `setld -l output`
6. During the install, if you intend to use the GUI you must select the optional GUI subset (QVETXOSFxxx).
7. The Q-Vet installation will size your system for devices and memory. It also runs qvet_tune. You should answer '**`y`**' to the questions that are asked about setting parameters. If you do not, you may have trouble running Q-Vet. After the installation completes, you should delete the output directory with **`rm -r output`**. You can also delete the kit tar file.
8. You ***must*** reboot the system before starting Q-Vet.
9. On reboot you can start Q-Vet GUI via **`vet&`** or you can run nonGUI (command line) via **`vet –nw`**.

OpenVMS

1. Delete any *QVETAXPxxx.A* or *QVETAXPxxx.EXE* file from the current directory.
2. Copy the self-extracting kit image file (*QVETAXPxxx.EXE*) to the current directory.
3. It is highly recommended, but not required, that you purge the system disk before installing Q-Vet. This will free up space that may be needed for pagefile expansion during the AUTOGEN phase.
   $purge sys$sysdevice:[*…]*.*
4. Extract the kit saveset with the command **`$run QVETAXPxxx.EXE`** and verify that the kit saveset was extracted by checking for the "Successful decompression" message.
5. Use **`@sys$update:vmsinstal`** for the Q-Vet installation. The installation will size your system for devices and memory. You should choose all the default answers during the Q-Vet installation. This will verify the Q-Vet installation, tune the system, and reboot. During the install, if you ***do not*** intend to use the GUI, you can answer **`no`** to the question "Do you want to install Q-Vet with the DECwindows Motif interface?"
6. After the installation completes you should delete the *QVETAXP0xx.A* file and the *QVETAXPxxx.EXE* file.
7. On reboot you can start Q-Vet GUI via **$vet** or the command interface via **$vet/int=char.**

2.6.2 Running Q-Vet

You must run Q-Vet on each partition in the system to verify the complete system.

Review the Special Notices and the Testing Notes section of the Release Notes located at http://cisweb.mro.cpqcorp.net/projects/qvet/ before running Q-Vet.

Follow the instructions listed for your operating system to run Q-Vet in each partition.

Tru64 UNIX

Graphical Interface	1. From the Main Menu, select **IVP**, **Load Script** and select **Long IVP** (the IVP tests will then load into the Q-Vet process window). 2. Click the **Start All** button to begin IVP testing.

Command-Line Interface	`> vet -nw`
	`Q-Vet_setup> execute .Ivp.scp`
	`Q-Vet_setup> start`
	
	Note that there is a "**.**" in front of the script name, and that commands are case sensitive.

OpenVMS

Graphical Interface

1. From the Main Menu, select `IVP`, **Load Script** and select **Long IVP** (the IVP tests will then load into the Q-Vet process window).
2. Click the **Start All** button to begin IVP testing.

Command-Line Interface

```
$ vet /int=char
Q-Vet_setup> execute ivp.vms
Q-Vet_setup> start
```

Note that commands are case sensitive.

NOTE: *A short IVP script is provided for a simple verification of device setup. It is selectable from the GUI IVP menu, and the script is called* ***.Ivp_short.scp*** *(**ivp_short.vms**). This script will run for 15 minutes and then terminate with a Summary log. The short script may be run prior to the long IVP script if desired, but not in place of the long IVP script, which is the full IVP test.*

The long IVP will run until the slowest device has completed one pass (typically 2 to 12 hours). This is called a Cycle of Testing.

2.6.3 Reviewing Results of the Q-Vet Run

After running Q-Vet, check the results of the run by reviewing the summary log.

If you follow the above steps, Q-Vet will run all exercisers until the slowest device has completed one full pass. Depending on the size of the system (number of CPUs and disks), this will typically take 2 to 12 hours. Q-Vet will then terminate testing and produce a summary log. The termination message will tell you the name and location of this file.

All exerciser processes can also be manually terminated with the Suspend and Terminate buttons (**stop** and **terminate** commands).

After all exercisers report “Idle,” the summary log is produced containing Q-Vet specific results and statuses.

- If there are no Q-Vet errors, no system event appendages, and testing ran to the specified completion time, the following message will be displayed:

  ```
  ”Q-Vet Tests Complete: Passed”
  ```

- Otherwise, a message will indicate:

  ```
  ”Additional information may be available from System Event
  Analyzer”
  ```

It is recommended that you run System Event Analyzer to review test results. The testing times (for use with System Event Analyzer) are printed to the Q-Vet run window and are available in the summary log.

2.6.4 De-Installing Q-Vet

The procedures for de-installation of Q-Vet differ between operating systems. You must de-install Q-Vet from each partition in the system. Failure to do so may result in the loss of customer data at a later date if Q-Vet is misused.

Follow the instructions listed under your operating system to de-install Q-Vet from a partition. The **qvet_uninstall** programs will remove the Q-Vet supplied tools and restore the original system tuning/configuration settings.

Tru64 UNIX

1. **Stop, Terminate,** and **Exit** from Q-Vet testing.
2. Execute the command **`qvet_uninstall.`** This will also restore the system configuration/tuning file `sysconfigtab`.
3. Note: log files are retained in `/usr/field/tool_logs`
4. Reboot the system. You must reboot in any case, even if Q-Vet is to be reinstalled.

OpenVMS

1. **Stop**, **Terminate,** and **Exit** from any Q-Vet testing.
2. Execute the command **@sys$manager:qvet_uninstall.** This will restore system tuning (modparams.dat) and the original UAF settings.
3. Note: log files are retained in **sys$specific:[sysmgr.tool_logs]**
4. Reboot the system. You must reboot in any case, even if Q-Vet is to be reinstalled.

2.7 Information Resources

Many information resources are available, including tools that can be downloaded from the Internet, firmware updates, a supported options list, and more.

2.7.1 HP Service Tools CD

The HP Service Tools CD-ROM enables field engineers to upgrade customer systems with the latest version of software when the customer does not have access to HP Web pages. The Web site is:
http://www.mse.qvar.cpqcorp.net/ServiceTools/default.asp

2.7.2 DS15 Service HTML Help File

The information contained in this guide, including the FRU procedures and illustrations, is available in HTML Help format as part of the Maintenance Kit. It can also be accessed from the Learning Utility and ProSIC Web sites.

2.7.3 Alpha Systems Firmware Updates

The firmware resides in the flash ROM on the system motherboard. You can obtain the latest system firmware from CD-ROM or over the network.

Quarterly Update Service

The Alpha Systems Firmware Update Kit CD-ROM is available by subscription.

Alpha Firmware Internet Access

- You can obtain Alpha firmware updates from the following Web site:
  http://ftp.digital.com/pub/Digital/Alpha/firmware/readme.html
- The README file describes the firmware directory structure and how to download and use the files.
- If you don't have a Web browser, download the files using anonymous ftp:

  ftp://ftp.digital.com/pub/Digital/Alpha/firmware/
- Individual Alpha system firmware releases that occur between releases of the firmware CD are located in the interim directory:
- ftp://ftp.digital.com/pub/Digital/Alpha/firmware/interim/

2.7.4 Fail-Safe Booter

The fail-safe booter (FSB) allows you to run another console to repair files that reside in the flash ROMs on the system motherboard. See Chapter 3 for information on running the FSB.

2.7.5 Software Patches

Software patches for the supported operating systems are available from:

http://h18002.www1.hp.com/alphaserver/

2.7.6 Learning Utility

The Learning Utility provides information about various technical topics.

http://learning1.americas.cpqcorp.net/mcsl-html/home.asp

2.7.7 Late-Breaking Technical Information

You can download up-to-date files and late-breaking technical information from the Internet.

The information includes firmware updates, the latest configuration utilities, software patches, lists of supported options, and more.

http://h18002.www1.hp.com/alphaserver/

2.7.8 Supported Options

A list of options supported on the system is available on the Internet:

http://h18002.www1.hp.com/alphaserver/

Chapter 3
Power-Up Diagnostics and Display

This chapter describes the power-up process and RMC, SROM, and SRM power-up diagnostics. The following topics are covered:

- Overview of Power-Up Diagnostics
- System Power-Up Sequence
- Power-Up Displays
- Power-Up Error Messages
- Forcing a Fail-Safe Load
- Updating the RMC

3.1 Overview of Power-Up Diagnostics

The power-up process begins with the power-on of the power supply. After the AC and DC power-up sequences are completed, the remote management console (RMC) reads EEROM information and deposits it into the dual-port RAM (DPR). The SROM minimally tests the CPU, initializes and tests backup cache, and minimally tests memory. Finally, the SROM loads the SRM console program into memory and jumps to the first instruction in the console program.

There are three distinct sets of power-up diagnostics:

1. System power controller and remote management console diagnostics – These diagnostics check the power regulators, temperature, and fans. Failures are reported in the dual-port RAM and on the Operator Control Panel (OCP) LEDs. Certain failures may prevent the system from powering on.
2. Serial ROM (SROM) diagnostics – SROM tests check the basic functionality of the system and load the console code from the FEPROM on the system motherboard into system memory. Failures during SROM tests are indicated by error beep codes and messages to the power-up console terminal.
3. Console firmware diagnostics – These tests are executed by the SRM console code. They test the core system, including boot path devices. Failures during these tests are reported to the console terminal through the power-up screen or console event log.

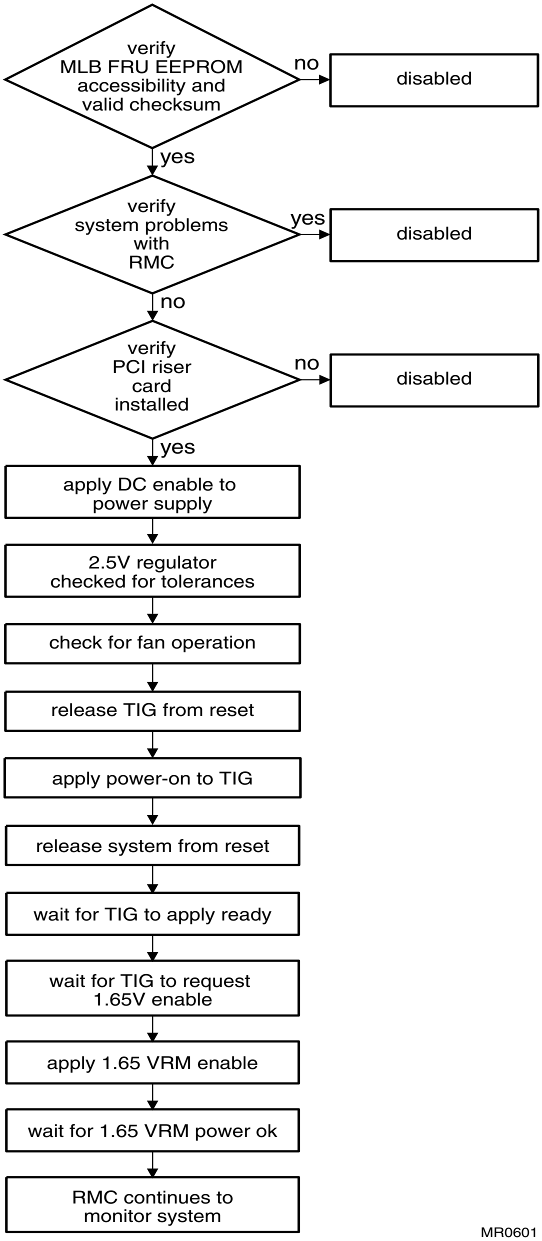
3.2 System Power-Up Sequence

The power-up sequence is described below and illustrated in Figure 3–1.

The RMC is responsible for the power-up sequence of the *AlphaServer* DS15. The general power-up sequence follows:

1. Verify that the MLB FRU EEPROM is accessible and has a valid checksum. The system will not be allowed to power-on unless these conditions are met (this check – and others - can be disabled with the FEATURE_0 jumper).
2. Verify that the RMC did not detect any system problems during its power-on self-test (this check can be overridden with the FEATURE_0 jumper).
3. Verify that the PCI Riser Card (PRC) is installed (this check can be disabled with the FEATURE_3 jumper).
4. Assert DC Enable to the bulk power supply and wait for Power OK (POK) to assert.
5. Check the 2.5V regulator to ensure it is within specified tolerances.
6. Check to see if the disk fan is spinning. If it is and the FEATURE_4 jumper is installed, flag a configuration error.
7. Release Titan Interrupt and General logic chip (TIG) from reset.
8. Assert power-on signal to TIG.
9. Release system from reset.
10. Wait for TIG to assert ready.
11. Wait for TIG to request 1.65V enable.
12. Assert the 1.65V VRM enable signal.
13. Wait for the 1.65V VRM Power OK signal to assert.

Figure 3–1 Power-Up Sequence



3.3 Power-Up Displays

Power-up information is displayed on the OCP LEDs and on the console terminal startup screen. Messages sent from the RMC and SROM programs are displayed first, followed by messages from the SRM console.

3.3.1 Power-Up Display

The following example describes the power-up sequence and shows the power-up messages.

Example 3–1 Sample Power-Up Display

```
SROM V1.0-0 CPU # 00 @  1000  MHz            PCI Test         ❶
SROM program starting                        Program on       ❷
Reloading SROM
............
SROM V1.0-1 CPU # 00 @  1000  MHz
 System Bus Speed @ 0125  MHz
SROM program starting                        RelCPU           ❸
Bcache data tests in progress
Bcache address test in progress
CPU parity and ECC detection in progress
Bcache ECC data tests in progress            BC Data          ❹
Bcache TAG lines tests in progress
Memory sizing in progress                    Size Mem         ❺
Memory configuration in progress
Testing AAR2
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory initialization
............Loading console                  Load ROM Jump    ❻
Code execution complete (transfer control)   to console
```

Power-Up Sequence

❶ When the system powers up, the SROM code is loaded into the I-cache (instruction cache) on the CPU. Minimum amount of hardware is verified including the EV6 and certain Titan related items.

The CPU attempts to access the PCI bus. If it cannot, either a hang or a failure occurs, and this is the only message displayed.

Clock speed is determined.

At this point the SROM checks a jumper to see if it needs to go to the mini-debugger or wait for the RMC to complete populating the DPR.

❷ The CPU interrogates the I^2C EEROM on the system board through shared RAM. The CPU determines the system configuration to jump to.

The CPU next checks the SROM checksum to determine the validity of the flash SROM sectors.

If flash SROM is invalid, the CPU reports the error and continues the execution of the SROM code. Memory is programmed and tested and SROM transfers execution to the console indicating in the DPR that the flash is BAD. Invalid flash SROM must be reprogrammed.

If flash SROM is good, the CPU programs appropriate registers with the values from the flash data and selects itself as the target CPU to be loaded. When the SROM is reloaded from flash, the system will be programmed with correct values and running at correct speed.

❸ The CPU initializes and tests the B-cache and memory, then loads the flash SROM code. At this point code execution begins from STEP 1 just as the on-chip SROM code. However a flag indicates that the CPU is running flash SROM and there is no need to re-load the flash on the second pass.

❹ The flash SROM performs B-cache tests. For example, the ECC data test verifies the detection logic for single- and double-bit errors.

❺ The CPU initiates all memory tests. The memory is tested for address and data errors for the first 32 MB of memory in each array. It also initializes all the "sized" memory in the system.

If a memory failure occurs, an error is reported. An untested memory array is assigned to address 0 and the failed memory array is de-assigned. Memory tests are rerun on the first 32 MB of memory in each remaining arrays. If all memory fails, the "No Memory Available" message is reported and the system halts.

The CPU validates that its external interrupts are functioning.

❻ If all memory passes, the CPU loads the console and transfers control to it.

NOTE: *The power-up text that is displayed on the screen depends on what kind of terminal is connected as the console terminal: VT or VGA.*

If the SRM ***console*** *environment variable is set to* ***serial****, the entire power-up display, consisting of the SROM and SRM power-up messages, is displayed on the VT terminal screen. If* ***console*** *is set to* ***graphics****, no SROM messages are displayed, and the SRM messages are delayed until VGA initialization has completed.*

3.3.2 Console Power-Up Display

When power-up is complete, the CPU transfers control to the SRM console. The console continues the system initialization. Failures are reported to the console terminal through the power-up screen and a console event log.

The following section shows the messages that are displayed once the SROM has transferred control to the SRM console.

Example 3–2 Power-Up Display

```
OpenVMS PALcode V1.98-6, Tru64 UNIX PALcode V1.92-7                    ❶
starting console on CPU 0
initialized idle PCB
initializing semaphores
initializing heap
initial heap 240c0
memory low limit = 1be000 heap = 240c0, 17fc0
initializing driver structures
initializing idle process PID
initializing file system
initializing timer data structures
lowering IPL
CPU 0 speed is 1000 MHz
create dead_eater
create poll
create timer
create powerup
access NVRAM
2048 MB of System Memory
Testing Memory                                                         ❷

...
probe I/O subsystem                                                    ❸
starting drivers                                                       ❹
```

❶ The primary CPU prints a message indicating that it is running the console. Starting with this message, the power-up display is sent to any console terminal, regardless of the state of the **console** environment variable.

If console is set to **graphics**, the display from this point on is saved in a memory buffer and displayed on the VGA monitor after the PCI buses are sized and the VGA device is initialized.

❷ The memory size is determined and memory is tested.

❸ The I/O subsystem is probed and I/O devices are reported. I/O adapters are configured.

❹ Device drivers are started.

Example 3–2 Power-Up Display (Continued)

```
entering idle loop                                                   ❺

initializing keyboard
initializing GCT/FRU at 1f0000
Initializing dqa dqb eia eib pka pkb
Memory Testing and Configuration Status                              ❻
  Array       Size       Base Address    Intlv Mode
---------  ----------  ----------------  ----------
    0       1024Mb     0000000000000000    2-Way
    2       1024Mb     0000000040000000    2-Way
     2048 MB of System Memory
Testing the System
Testing the Disks (read only)
Testing the Network
AlphaServer DS15 Console X6.6-3090, built on Aug 14 2003 at 00:42:53   ❼
>>>
```

❺ Entering the idle loop.

❻ Various diagnostics are performed.

❼ The console terminal displays the SRM console banner and the prompt, >>>. From the SRM prompt, you can boot the operating system.

NOTE: *If the console requires the heap to be expanded, it restarts.*

3.3.3 SRM Console Event Log

The SRM console event log helps you troubleshoot problems that do not prevent the system from coming up to the SRM console. The console event log consists of status messages received during power-up self-tests.

Example 3–3 Sample Console Event Log

```
> >>>more el
> starting console on CPU 0
> initialized idle PCB
> initializing semaphores
> initializing heap
> initial heap 240c0
> memory low limit = 1be000 heap = 240c0, 17fc0
> initializing driver structures
> initializing idle process PID
> initializing file system
> initializing timer data structures
> lowering IPL
> CPU 0 speed is 1000 MHz
> create dead_eater
> create poll
> create timer
> create powerup
> access NVRAM
> Testing Memory
> ...
> probe I/O subsystem
> starting drivers
> entering idle loop
> initializing keyboard
> --More-- (SPACE - next page, ENTER - next line, Q - quit)
> port dqa.0.0.13.0 initialized
> port dqb.0.1.13.0 initialized
> device dqa0.0.0.13.0 (DW-224E) found on dqa0.0.0.13.0
> device dka0.0.0.8.0 (COMPAQ BF03665A32) found on pka0.0.0.8.0
> device dka100.1.0.8.0 (COMPAQ BF03665A32) found on pka0.1.0.8.0
> sense key = 'Unit Attention' (29|02) from dka0.0.0.8.0
> Change to Internal loopback.
> Change to Normal Operating Mode.
> Change to Internal loopback.
> Change to Normal Operating Mode.
> >>>
```

To check for and locate errors, enter the **Log** command at the RMC> prompt.

Example 3–4 Using the Log Command to Check for Errors

```
RMC>log
Entry 00: Fan failure
Total Entries = 1
RMC>log 00
Event Log Entry 0
Primary Event: Fan failure
Secondary Messages:
PCI fan speed failure
RMC initiated delayed system shutdown
Voltages:
1.65V    :    1.66V    2.5V     :     2.49V      3.3V Bulk :   3.37V
5V Bulk  :    5.14V    12V Bulk  :   12.24V    -12V Bulk :   -12.19V
3.3Vsb   :    3.30V    5Vsb Bulk :    5.04V      2.85V (A) :   2.83V
2.85V (B) :   2.85V
Temperature:
Inlet Air :    26.000C
Fans:
System Fan:   2010RPM   PCI Fan   :    0RPM   CPU Fan   :   3450RPM
Disk Fan  :     2730RPM
Shutdown Overrides:
Thermal Shutdown: Enabled
Fan Shutdown: Enabled
RMC>
```

3.4 Power-Up Error Messages

Audible beep codes announce specific errors that might be encountered while the system is powering up. Table 3–1 identifies the error beep codes.

Table 3–1 Error Beep Codes

Beeps	Message/Meaning	Action to Repair
1	Done with execution; jumping to console	No action necessary.
1-3-3	No usable memory available	Check memory and memory configuration.
2-1-2	Configuration error detected	Check system configurations.
1-1-4	ROM checksum error detected	Replace the system board.
1-2-4	Bcache error detected	Possible CPU problem.

3.4.1 Checksum Error

When the system detects the error, it attempts to load the fail-safe booter (FSB) console so that you can load new console firmware images. A sequence similar to the one in Example 3–5 occurs.

Example 3–5 Checksum Error and Fail-Safe Boot Console

```
SROM V1.0-0 CPU # 00 @  1000  MHz
SROM program starting
Reloading SROM
............
 SROM V1.0-1 CPU # 00 @  1000  MHz
 System Bus Speed @ 0125  MHz
SROM program starting
Bcache data tests in progress
Bcache address test in progress
CPU parity and ECC detection in progress                              ❶
Bcache ECC data tests in progress
Bcache TAG lines tests in progress
Memory sizing in progress
Memory configuration in progress
Testing AAR2
Memory data test in progress                                          ❷
Memory address test in progress
Memory pattern test in progress
Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory initialization
............Loading console
Expect: 00000000.000000F1
Actual: 00000000.00000075
XORval: 00000000.00000084
loading program from floppy
Floppy driver error
Loading Fail-Safe console
Code execution complete (transfer control)

OpenVMS PALcode V1.98-6, Tru64 UNIX PALcode V1.92-7


starting console on CPU 0                                             ❸
initialized idle PCB
initializing semaphores
initializing heap
initial heap 240c0
memory low limit = 1a0000 heap = 240c0, 17fc0
initializing driver structures
initializing idle process PID
initializing file system
initializing timer data structures
lowering IPL
CPU 0 speed is 1000 MHz
create dead_eater
```

```
create poll
create timer
create powerup
access NVRAM
1024 MB of System Memory
Testing Memory
...
probe I/O subsystem
starting drivers
entering idle loop
initializing keyboard
initializing GCT/FRU at 1cc000
Initializing dqa dqb eia eib pka pkb


***************************************************************************
*                                                                         *
*                DS15 Failsafe Boot Console                               *
*Please use the LFU utility to update/recover your SRM console flash image.*
*                                                                         *
***************************************************************************
AlphaServer DS15 Console X6.6-3140, built on Aug 15 2003 at 00:53:42
>>>                                                                       ❹
```

The sequence shown in Example 3–5 is as follows:

❶ ECC detection in progress.

❷ Memory data test in progress.

❸ As the FSB console is initialized, messages similar to the console power-up messages are displayed. This example shows the beginning and ending messages.

❹ At the >>> console prompt, boot the Loadable Firmware Update Utility (LFU) from the Alpha Systems Firmware CD .

NOTE: *For more information on LFU, see the Firmware Updates Web site:* *http://ftp.digital.com/pub/digital/Alpha/firmware/*

3.4.2 SROM Memory Configuration Errors

If the SROM fails, the display will show all the DIMMs that are missing. The system uses the JEDEC data on the DIMM and reports configuration errors if no memory is available for the console.

The system reports DIMM failure as ILLEGAL, MISSING, INCMPAT, or FAILED. The following excerpts are examples of error reports. See Chapter 6 for memory configuration rules.

Example 3–6 Report for Illegal DIMM

```
Report for Illegal DIMM
*** Data Detected Memory Error ***
Memory sizing in progress
Memory configuration in progress
Testing AAR2
Memory data test in progress
Memory data path error

ErrAdr: 00000000.00000000
Expect: 00000000.00000001
Actual: 00000000.00000000
XORval: 00000000.00000001

Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory initialization
Failed  DIMM 3
Loading console
Code execution complete (transfer control)
*****************************************
```

Example 3–7 Report for Missing DIMM

```
*** Missing DIMM Error ***
Testing AAR2
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
```

```
Memory initialization
Missing  DIMM 1
Loading console
Code execution complete (transfer control)
***********************************
```

Example 3–8 Report for Incompatible DIMM

```
*** Incompatible dimm error ***
Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory initialization
Incmpat  DIMM 2
Incmpat  DIMM 0

Loading console
*********************************
```

Example 3–9 Report for Failed DIMM

```
*** ECC detected error ***
Memory sizing in progress
Memory configuration in progress
Testing AAR2
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory pattern ECC error

Expect: 00000000.00000000
Actual: 00000000.0000000C
XORval: 00000000.0000000C
C0_SYN: 00000000.00000008
C1_SYN: 00000000.00000000
C_STS:  00000000.00000000
C_STAT: 00000000.00000003
C_ADDR: 00000000.20000000
D_STAT: 00000000.0000000C

Testing AAR0
Memory data test in progress
Memory address test in progress
Memory pattern test in progress
Memory initialization
*********************************
```

3.5 Forcing a Fail-Safe Load

The fail-safe booter is another variant of the SRM console. The FSB provides an emergency recovery mechanism if the firmware image contained in flash memory becomes corrupted. You can run the FSB and boot another image from a CD-ROM or network that is capable of reprogramming the flash ROM.

Use the FSB when one of the following failures at power-up prohibits you from getting to the console program:

- Firmware image in flash memory corrupted
- Power failure or accidental power-down during a firmware upgrade
- Error in the nonvolatile RAM (NVRAM) file
- Incorrect environment variable setting
- Driver error

3.5.1 Starting the FSB Automatically

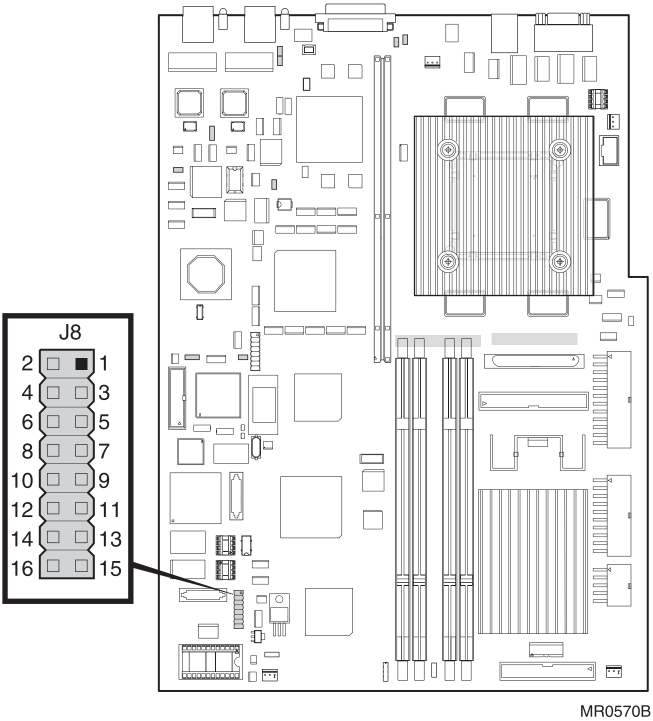
If the firmware image is unavailable when the system is powered on or reset, the FSB runs automatically.

1. Reset the system to restart the FSB. The FSB loads from the flash.
2. Update the firmware as described in Chapter 7.

3.5.2 Starting the FSB Manually

1. Power the system off, unplug the AC power cord, and remove the top cover. (See Chapter 8 for instructions.)
2. Insert jumper J8 over pins 1-2 on the system motherboard. See Figure 3–2.
3. Reconnect the AC power cord and reinstall the system cover. Power up the system to the FSB console.

Figure 3–2 FSB Switch "On" Setting (Rackmounted Orientation)



3.6 Updating the RMC

Under certain circumstances, the RMC will not function. If the problem is caused by corrupted RMC flash ROM, you need to update RMC firmware.

The RMC will not function if:

- No AC power is provided.
- DPR does not pass its self-test (DPR is corrupted).
- RMC flash ROM is corrupted, but the system will still power-up.

The SRM console also sends a message to the terminal screen:

```
*** Error - RMC detected power up error - RMC Flash corrupted ***
```

You can update the remote management console firmware from flash ROM using the LFU. For details, see Chapter 7, RMC Firmware Update and Recovery.

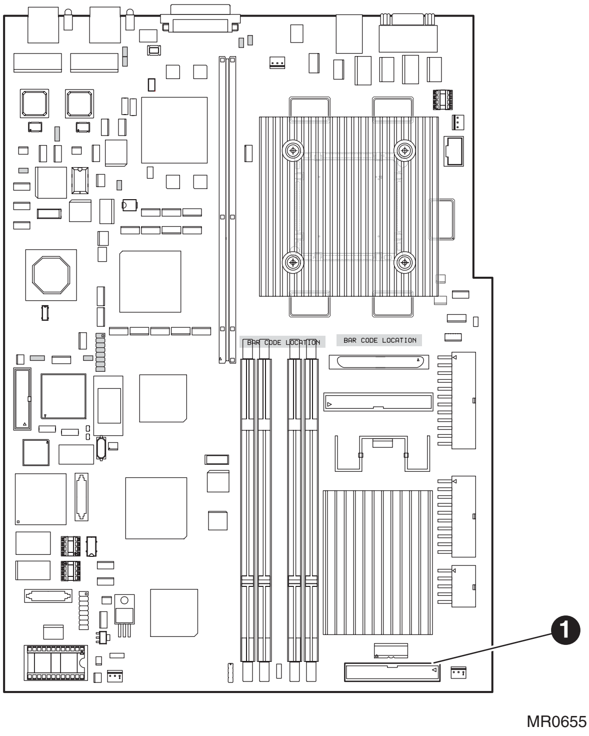
NOTE: *For more information on LFU, see the Firmware Updates Web site:* *http://ftp.digital.com/pub/digital/Alpha/firmware/*

3.7 Field Use of a Floppy Diskette

The DS15 does not ship with a floppy diskette device. However, the console software and hardware maintain floppy support. Carrying a floppy device and associated cabling could be quite handy in the field if there are no other means to update the console firmware. Additionally, if a motherboard needs to be replaced, one can preserve the customer NVRAM settings by invoking the save_nvram and restore_nvram console commands.

The floppy device plugs into the motherboard at connector J4, as shown in the following figure.

Figure 3–3 Location of Floppy Device Connector



Chapter 4
SRM Console Diagnostics

This chapter describes troubleshooting with the SRM console.

The SRM console firmware contains ROM-based diagnostics that allow you to run system-specific or device-specific exercisers. The exercisers run concurrently to provide maximum bus interaction between the console drivers and the target devices.

Run the diagnostics by using commands from the SRM console. To run the diagnostics in the background, use the background operator “&” at the end of the command. Errors are reported to the console terminal, the console event log, or both.

If you are not familiar with the SRM console, see the *AlphaServer DS15 and AlphaStation DS15 Owner’s Guide*.

4.1 Diagnostic Command Summary

Diagnostic commands are used to test the system and help diagnose failures. Table 4–1 gives a summary of the SRM diagnostic commands and related commands. See Chapter 6 for a list of SRM environment variables, and see Chapter 7 for a list of RMC commands most commonly used for the DS15 system.

Table 4–1 Summary of Diagnostic and Related Commands

Command	Function
buildfru	Initializes I^2Cbus EEPROM data structures for the named FRU.
cat el	Displays the console event log. Same as **more el**, but scrolls rapidly. The most recent errors are at the end of the event log and are visible on the terminal screen.
clear_error	Clear errors logged in the FRU EEPROMs as reported by the **show error** command.
crash	Forces a crash dump at the operating system level.
deposit	Writes data to the specified address of a memory location, register, or device.
examine	Displays the contents of a memory location, register, or device.
exer	Exercises one or more devices by performing specified read, write, and compare operations.
grep	Searches for “regular expressions” – specific strings of characters – and prints any lines containing occurrences of the strings.
hd	Dumps the contents of a file (byte stream) in hexadecimal and ASCII.
info	Displays registers and data structures.
kill	Terminates a specified process.
kill_diags	Terminates all executing diagnostics.

Continued on next page

Table 4–1 Summary of Diagnostic and Related Commands (Cont'd.)

Command	Function
more el	Same as **cat el**, but displays the console event log one screen at a time.
memexer	Runs a requested number of memory tests in the background.
memtest	Tests a specified section of memory.
net -ic	Initializes the MOP counters for the specified Ethernet port.
net -s	Displays the MOP counters for the specified Ethernet port.
nettest	Runs loopback tests for PCI-based Ethernet ports. Also used to test a port on a "live" network.
scsi_poll	Controls whether or not a particular SCSI device driver polls for devices on the bus when the driver is started. This device is supported by some, but not all, console SCSI device drivers.
scsi_reset	Controls whether or not a particular SCSI device driver resets the SCSI bus when the driver is started. This EV is supported by some, but not all, console SCSI device drivers.
set sys_serial_ num	Sets the system serial number, which is then propagated to all FRUs that have EEPROMs.
sys_com1_rmc	Enables/disables internal COM1 access to the RMC.
show error	Reports errors logged in the FRU EEPROMs.
show fru	Displays information about field replaceable units (FRUs), including CPUs, memory DIMMs, and PCI cards.
show_status	Displays the progress of diagnostic tests. Reports one line of information for each executing diagnostic.
sys_exer	Exercises the devices displayed with the **show config** command.
sys_exer -lb	Runs console loopback tests for the COM2 serial port during the **sys_exer** test sequence.
test	Verifies the configuration of the devices in the system.
test -lb	Runs loopback tests for the COM2 serial port in addition to verifying the configuration of devices.

4.2 Buildfru

The buildfru command initializes I^2C bus EEPROM descriptive data structures for the named FRU and initializes its SDD and TDD error logs. This command uses data supplied on the command line to build the FRU descriptor. Buildfru is used by Manufacturing, FRU repair operations, or Field Service.

The **buildfru** command is used for several purposes:

- By Manufacturing to build a FRU table containing a description of each FRU in the system.
- By FRU repair operations for initializing good stocking spares
- By Field Service to make any FRU descriptor adjustments required by the customer.

Example 4–1 Buildfru Command

1. Pass the motherboard part and serial number.

```
>>> buildfru HMB 54-30558-b01 sw12345678
```

2. Build the motherboard EEPROM at offset 80 with value of 45

NOTE: *Use the -s option with care. It could corrupt the FRU EPROM and render the system unusable. Address and values are in hexadecimal.*

```
>>> buildfru -s hmb 80 45
```

3. Build the motherboard EEPROM at offset 80 with sequential data: the value at offset 80 is 47, the value at offset 81 is 46, and so on.

```
>>> buildfru -s hmb 80 47 46 45 44 43 42 41
```

The information supplied on the **buildfru** command line includes the console name for the FRU, part number, serial number, model number, and optional information. The **buildfru** command facilitates writing the FRU information to the EEPROM on the device.

Use the **show fru** command to display the FRU table created with **buildfru**. Use the **show error** command to display FRUs that have errors logged to them.

Typically, you only need to use **buildfru** in Field Service if you replace a device for which the information displayed with the **show fru** command is inaccurate or missing. After replacing the device, use **buildfru** to build the new FRU descriptor.

NOTE: *Be sure to enter the FRU information carefully. If you enter incorrect information, the callout used by System Event Analyzer will not be accurate.*

Three areas of the EEPROM can be initialized: the FRU generic data, the FRU specific data, and the system specific data. Each area has its own checksum, which is recalculated any time that segment of the EEPROM is written.

When the **buildfru** command is executed, the FRU EEPROM is first flooded with zeros and then the generic data, the system specific data, and EEPROM format version information are written and checksums are updated. For certain FRUs, such as CPU modules, additional FRU "specific" data can be entered using the **-s** option. This data is written to the appropriate region, and its corresponding checksum is updated.

FRU Assembly Hierarchy

Alpha-based systems can be decomposed into a collection of FRUs. Some FRUs carry various levels of nested FRUs. For instance, the system motherboard is a FRU that carries a number of "child" FRUs such as DIMMs. The naming convention for FRUs represents the assembly hierarchy.

The following is the general form of a FRU name:

<fru*n*>[.<fru*n*>[.<fru*n*>]]

The *fru* is a placeholder for the appropriate FRU type at that level and *n* is the number of that FRU instance on that branch of the system hierarchy.

The DS15 FRU assembly hierarchy has three levels. The FRU types from the top to the bottom of the hierarchy are as follows:

Level	FRU Type	Description
First Level	HMB PWR0 FAN	System motherboard Power supply Fans for system, PCI, disks, CPU
Second Level	HMB.DIMM(0-3) HMB.PCIRSR	Memory DIMMs PCI riser card
Third Level	HMB.PCIRSR.PCI	PCI slots (1-4)

To build an FRU descriptor for a lower-level FRU, point back to the higher-level FRUs to which the lower-level FRU is associated. See preceding Section 4.2.

If you enter the **buildfru** data correctly for a device that has an EEPROM to program, nothing is displayed after you enter the command. If you enter incorrect data or the device does not have an EEPROM to program, an error message similar to the following is displayed:

```
>>>buildfru "sys fan" 12-10010-01 ay12345678
Device SYS FAN does not support setting FRU values
>>>
```

Syntax

buildfru (<fru_name> <part_num> <serial_num> [<misc> [<other>]]
or
-s <fru_name> <offset> <byte> [<byte>...])

Arguments

<fru_name>	Console name for this FRU. This name reflects the position of the FRU in the assembly hierarchy.
<part_num>	The FRU 2-5-2.4 part number. This ASCII string should be 16 characters (extra characters are truncated). This field should not contain any embedded spaces. If a space must be inserted, enclose the entire argument string in double quotes. This field contains the FRU revision. In some cases, an embedded space is allowed between the part number and the revision.
<serial_num>	The FRU serial number. This ASCII string must be 10 characters (extra characters are truncated). The manufacturing location and date are extracted from this field.
<misc>	The FRU model name, number, or the common name for the FRU. This ASCII string may be up to 10 characters (extra characters are truncated). This field is optional, unless **<alias>** is specified.
<other>	The FRU HP alias number, if one exists. This ASCII string may be up to 16 characters (extras are truncated). This field is optional.
<offset>	The beginning byte offset (0–255 hex) within this FRU EEPROM, where the following supplied data bytes are to be written.
<byte>...	The data bytes to be written. At least one data byte must be supplied after the offset.

Options

-s	Writes raw data to the EEPROM. This option is typically used to apply any FRU specific data.
-dimm	Generates a unique serial number for each DIMM in the system.

4.3 cat el and more el

The cat el and more el commands display the contents of the console event log.

In Example 4–2, the console reports that the CPU did not power up and fans 1 and 2 failed.

Example 4–2 more el

```
>>> more el
*** Error - CPU  failed powerup diagnostics ***
  Secondary start error
EV6 BIST         = 1
STR status       = 1
CSC status       = 1
PChip0 status    = 1
DIMx status      = 0
TIG Bus status   = 1
DPR status       = 0
CPU speed status = ff
CPU speed        = 1000
Powerup time     = 08-06-51 14:30:19
CPU SROM sync    = 0

DPR has failed.  1=good 0=bad
```

Status and error messages are logged to the console event log at power-up, during normal system operation, and while running system tests. Standard error messages are indicated by asterisks (***).

When **cat el** is used, the contents of the console event log scroll by. Use the Ctrl/S key combination to stop the screen from scrolling, and use Ctrl/Q to resume scrolling.

The **more el** command allows you to view the console event log one screen at a time.

Syntax

cat el or **more el**

4.4 clear_error

The clear_error command clears errors logged in the FRU EEPROMs as reported by the show error command.

Example 4–3 clear_error

```
>>>clear_error HMB          ❶
>>>

>>>clear_error all          ❷
>>>
```

❶ Clears all errors logged in the FRU EEPROM on the system motherboard (HMB).

❷ Clears all errors logged to all FRU EEPROMs in the system.

The **clear_error** command clears TDD, SDD, and checksum errors. Hardware failures and unreadable EEPROM errors are not cleared. See Table 4–2.

Syntax

clear_error <*fruname*>	Clears all errors logged to a specific FRU. *Fruname* is the name of the specified FRU. If you do not specify a FRU, you must use **clear_error all** to clear errors.
clear_error all	Clears all errors logged to all system FRUs.

See the **show error** command for information on the types of errors that might be logged to the FRU EEPROMs.

4.5 crash

The SRM crash command forces a crash dump to the selected device for *Tru64 UNIX* and *OpenVMS* systems.

```
>>>crash
CPU 0 restarting

operator requested crash dump on cpu 0
DUMP: blocks available: 66044768
DUMP: blocks wanted:      168562 (partial compressed dump) [OKAY]
DUMP: Device      Disk Blocks Available
DUMP: ------      ---------------------
DUMP: 0x1300003  449308 - 786429 (of 786430) [primary swap]
DUMP.prom: Open: dev 0x5100001, block 786432: SCSI 0 8 0 0 0 0 0
DUMP: Writing header... [1024 bytes at dev 0x1300003, block 786430]
DUMP: Writing data....... [7MB]
DUMP: Writing header... [1024 bytes at dev 0x1300003, block 786430]
DUMP: crash dump complete.
halted CPU 0
halt code = 5
HALT instruction executed
PC = fffffc00008f0aac
>>>
```

Use the **crash** command when the system has hung and you are able to halt it with the halt/reset button (if configured for halt) or the RMC **halt** command. The **crash** command restarts the operating system and forces a crash dump to the selected device.

- See the *OpenVMS Alpha System Dump Analyzer Utility Manual* for information on how to interpret *OpenVMS* crash dump files.
- See the *Guide to Kernel Debugging* for information on using the *Tru64 UNIX* Krash Utility.

4.6 deposit and examine

The deposit command writes data to the specified address of a memory location, register, or device. The examine command displays the contents of a memory location, register, or a device.

Example 4–4 deposit and examine

Deposit

```
>>>dep -b -n 1ff pmem:0 0              ❶
>>>d -l -n 3 vmem:1234 5               ❷
>>>d -n 8 r0 ffffffff                  ❸
>>>d -l -n 10 -s 200 pmem:0 0          ❹
>>>d -l pmem:0 0                       ❺
>>>d + ff                              ❻
>>>d scbb 820000                       ❼
```

```
>>>e dpr:34f0 -l -n 5                  ❶
dpr:             34F0 00000000
dpr:             34F4 00000000
dpr:             34F8 00000000
dpr:             34FC 00000000
dpr:             3500 00000000
dpr:             3504 00000000
>>>
```

Deposit

The **deposit** command stores data in the location specified. If no options are given, the system uses the options from the preceding **deposit** command.

If the specified value is too large to fit in the data size listed, the console ignores the command and issues an error. If the data is smaller than the data size, the higher order bits are filled with zeros.

In Example 4–4:

❶ Clear first 512 bytes of physical memory

❷ Deposit 5 into four longwords starting at virtual memory address 1234.

❸ Load GPRs R0 through R8 with -1.

❹ Deposit 8 in the first longword of the first 17 pages in physical memory.

❺ Deposit 0 to physical memory address 0.

❻ Deposit FF to physical memory address 4.

❼ Deposit 820000 to SCBB.

Examine

The **examine** command displays the contents of a memory location, a register, or a device.

If no options are given, the system uses the options from the preceding **examine** command. If conflicting address space or data sizes are specified, the console ignores the command and issues an error.

For data lengths longer than a longword, each longword of data should be separated by a space.

In Example 4–4:

❶ Examine the DPR starting at location 34f0 and continuing through the next 5 locations, and display the data size in longwords.

Syntax

deposit [-{b,w,l,q,o,h}] [-{n *value, s value*}] [*space*:] *address data*

examine [-{b,w,l,q,o,h}] [-{n *value, s value*}] [*space:*] *address*

-b	Defines data size as byte.

-w	Defines data size as word.
-l (default)	Defines data size as longword.
-q	Defines data size as quadword.
-o	Defines data size as octaword.
-h	Defines data size as hexword.
-d	Instruction decode (examine command only)
-n *value*	The number of consecutive locations to modify.
-s *value*	The address increment size. The default is the data size.
dev_name	Device name (address space) of the device to access. Device names are:

dpr	Dual-port RAM. See Appendix C for the DPR address layout.
eerom	Nonvolatile ROM used for EV storage.
fpr	Floating-point register set; name is F0 to F31. Alternatively, can be referenced by name.
gpr	General register set; name is R0 to R31. Alternatively, can be referenced by name.
ipr	Internal processor registers. Alternatively, some IPRs can be referenced by name.
pcicfg	PCI configuration space.
pciio	PCI I/O space.
pcimem	PCI memory space
pt	The PALtemp register set; name is PT0 to PT23.
pmem	Physical memory (default).
vmem	Virtual memory.

offset	Offset within a device to which data is deposited.
data	Data to be deposited.

Symbolic forms can be used for the address. They are:

pc The program counter. The address space is set to GPR.

+ The location immediately following the last location referenced in a **deposit** or **examine** command. For physical and virtual memory, the referenced location is the last location plus the size of the reference (1 for byte, 2 for word, 4 for longword). For other address spaces, the address is the last referenced address plus 1.

- The location immediately preceding the last location referenced in a **deposit** or **examine** command. Memory and other address spaces are handled as above.

***** The last location referenced in a **deposit** or **examine** command.

@ The location addressed by the last location referenced in a **deposit** or **examine** command.

4.7 exer

The exer command exercises one or more devices by performing specified read, write, and compare operations. Typically exer is run from the built-in console script. Advanced users may want to use the specific options described here. Note that running exer on disks can be destructive.

Optionally, **exer** reports performance statistics:

- A read operation reads from a device that you specify into a buffer.
- A write operation writes from a buffer to a device that you specify.
- A compare operation compares the contents of the two buffers.

The **exer** command uses two buffers, buffer1 and buffer2, to carry out the operations. A read or write operation can be performed using either buffer. A compare operation uses both buffers.

Example 4–5 exer

```
>>> exer dk*.* -p 0 -secs 36000
```

Read SCSI disks for the entire length of each disk. Repeat this until 36000 seconds, 10 hours, have elapsed. All disks will be read concurrently. Each block read will occur at a random block number on each disk.

```
>>> exer -l 2 dka0
```

Read block numbers 0 and 1 from device dka0.

```
>>> exer -sb 1 -eb 3 -bc 4 -a 'w' -d1 '0x5a' dka0
```

Write hex 5a's to every byte of blocks 1, 2, and 3. The packet size is bc * bs, 4 * 512, 2048 for all writes.

```
>>>ls -l dk*.*
r---  dk                0/0                   0     0
dka0.0.0.8.0
r---  dk                0/0                   0     0
dka100.1.0.8.0

>>>exer dk*.* -bc 10 -sec 20 -m -a 'r'


dka0.0.0.8.0 exer completed
dka100.1.0.8.0 exer completed

packet                                             IOs
elapsed  idle
 size          IOs     bytes read  bytes written   /sec bytes/sec
seconds  secs
  8192       12753      104472576              0    635   5204632
20     15
>>>
```

A destructive write test over block numbers 0 through 100 on disk dka0. The packet size is 2048 bytes. The action string specifies the following sequence of operations:

1. Set the current block address to a random block number on the disk between 0 and 97. A four block packet starting at block numbers 98, 99, or 100 would access blocks beyond the end of the length to be processed so 97 is the largest possible starting block address of a packet.
2. Write a packet of hex 5a's from buffer1 to the current block address.
3. Set the current block address to what it was just prior to the previous write operation.
4. From the current block address read a packet into buffer2.
5. Compare buffer1 with buffer2 and report any discrepancies.
6. Repeat steps 1 through 5 until enough packets have been written to satisfy the length requirement of 101 blocks.

```
>>> exer –a '?r–w–Rc' dka0
```

A nondestructive write test with packet sizes of 512 bytes. Use this test only if the customer has a current backup of any disks being tested. The action string specifies the following sequence of operations:

1. Set the current block address to a random block number on the disk.
2. From the current block address on the disk, read a packet into buffer1.

3. Set the current block address to the device address where it was just before the previous read operation occurred.
4. Write the contents of buffer1 back to the current block address.
5. Set the current block address to what it was just prior to the previous write operation.
6. From the current block address on the disk, read a packet into buffer2.
7. Compare buffer1 with buffer2 and report any discrepancies.
8. Repeat the above steps until each block on the disk has been written once and read twice.

You can tailor the behavior of **exer** by using options to specify the following:

- An address range to test within the test device(s)
- The packet size, also known as the I/O size, which is the number of bytes read or written in one I/O operation
- The number of passes to run
- How many seconds to run
- A sequence of individual operations performed on the test devices. The qualifier is called the action string qualifier.

Syntax

exer ([-sb start_block>] [-eb end_block>] [-p pass_count>]
[-l blocks>] [-bs block_size>] [-bc block_per_io>] [-d1 buf1_string>]
[-d2 buf2_string>] [-a action_string>]
[-sec seconds>] [-m] [-v] [-delay milliseconds>] device_name>...)

Arguments	
device_name	Specifies the names of the devices or filestreams to be exercised.
Options	
-sb <start_block>	Specifies the starting block number (hex) within a filestream. The default is 0.
-eb <end_block>	Specifies the ending block number (hex) within filestream. The default is 0.
-p <pass_count>	Specifies the number of passes to run the exerciser. If 0, then run forever or until Ctrl/C. The default is 1.
-l <blocks>	Specifies the number of blocks (hex) to exercise. **-l** has precedence over **-eb**. If only reading, then specifying neither **-l** nor **-eb** defaults to read till end of file (eof). If writing, and neither **-l** nor **-eb** are specified then exer will write for the size of device. The default is 1.

-bs <block_size> Specifies the block size (hex) in bytes. The default is 200 (hex).

-bc <block_per_io> Specifies the number of blocks (hex) per I/O. On devices without length (tape), use the specified packet size or default to 2048. The maximum block size allowed with variable length block reads is 2048 bytes. The default is 1.

-d1 <buf1_string> String argument for eval to generate buffer1 data pattern from. Buffer1 is initialized only once before any I/O occurs. Default = all bytes set to hex 5A's.

-d2 <buf2_string> String argument for eval to generate buffer2 data pattern from. Buffer2 is initialized only once before any I/O occurs. Default = all bytes set to hex 5A's.

-a <action_string> Specifies an exerciser action string, which determines the sequence of reads, writes, and compares to various buffers. The default action string is ?r. The action string characters are:

- r Read into buffer1.
- W Write from buffer1.
- R Read into buffer2.
- W Write from buffer2.
- N Write without lock from buffer1.
- N Write without lock from buffer2.
- c Compare buffer1 with buffer2.
- \- Seek to file offset prior to last read or write.

-a <action_string> (continued)	• ? Seek to a random block offset within the specified range of blocks. **exer** calls the program, random, to "deal" each of a set of numbers once. **exer** chooses a set that is a power of two and is greater than or equal to the block range. Each call to random results in a number that is then mapped to the set of numbers that are in the block range and **exer** seeks to that location in the filestream. Since **exer** starts with the same random number seed, the set of random numbers generated will always be over the same set of block range numbers. • s Sleep for a number of milliseconds specified by the delay qualifier. If no delay qualifier is present, sleep for 1 millisecond. Times as reported in verbose mode will not necessarily be accurate when this action character is used. • z Zero buffer 1 • Z Zero buffer 2 • b Add constant to buffer 1 • B Add constant to buffer 2
-sec <seconds>	Specifies to terminate the exercise after the number of seconds have elapsed. By default the exerciser continues until the specified number of blocks or passcount are processed.
-m	Specifies metrics mode. At the end of the exerciser a total throughput line is displayed.
-v	Specifies verbose mode. Data read is also written to stdout. This is not applicable on writes or compares. The default is verbose mode **off**.
-delay <millisecs>	Specifies the number of milliseconds to delay when **s** appears as a character in the action string.

4.8 grep

The grep command is very similar to the UNIX grep command. It allows you to search for "regular expressions"—specific strings of characters—and prints any lines containing occurrences of the strings. Using grep is similar to using wildcards.

Example 4–6 grep

```
>>>sh fru | grep PCI
HMB.PCIRSR       00 54-30560-01.A1    SW31200018      -

HMB.PCIRSR.PCI  00                                    3D Labs OX
PCI FAN          00 12-49806-04                       FAN         J1
>>>
```

In Example 4–6 the output of the **show fru** command is piped into **grep** (the vertical bar is the piping symbol), which filters out only lines with "PCI."

Grep supports the following metacharacters:

^	Matches beginning of line
$	Matches end of line
.	Matches any single character
[ ]	Set of characters; [ABC] matches either 'A' or 'B' or 'C'; a dash (other than first or last of the set) denotes a range of characters: [A-Z] matches any uppercase letter; if the first character of the set is '^' then the sense of match is reversed: [^0-9] matches any non-digit; several characters need to be quoted with backslash (\) if they occur in a set: '\', ']', '-', and '^'
*****	Repeated matching; when placed after a pattern, indicates that the pattern should match any number of times. For example, '[a-z][0-9]*' matches a lowercase letter followed by zero or more digits.
+	Repeated matching; when placed after a pattern, indicates that the pattern should match one or more times '[0-9]+' matches any non-empty sequence of digits.
?	Optional matching; indicates that the pattern can match zero or one times. '[a-z][0-9]?' matches lowercase letter alone or followed by a single digit.
\	Quote character; prevent the character that follows from having special meaning.

Syntax

grep ([-{c|i|n|v}] [-f <file>] [<expression>] [<file>...])

Arguments

<expression> Specifies the target regular expression. If any regular expression metacharacters are present, the expression should be enclosed with quotes to avoid interpretation by the shell.

<file>... Specifies the files to be searched. If none are present, then standard input is searched.

Options

-c Print only the number of lines matched.

-i Ignore case. By default **grep** is case sensitive.

-n Print the line numbers of the matching lines.

-v Print all lines that do not contain the expression.

-f <file> Take regular expressions from a file, instead of command.

4.9 hd

The hd command dumps the contents of a file (byte stream) in hexadecimal and ASCII.

Example 4–7 hd

```
>>> hd -eb 0 dpr:2b00                                                      ❶

block 0
00000000  01 80 01 01 01 00 01 01 DD 01 FF E8 03 00 00 00   ........Ý..è....
00000010  17 53 43 31 07 51 00 7D 00 00 00 00 00 00 80 02   .SC1.Q.}........
00000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000080  40 10 00 00 41 10 00 00 00 00 00 00 00 00 00 00   @...A...........
00000090  00 01 00 00 02 03 00 00 02 08 00 00 00 00 00 00   ................
000000a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000000b0  00 C3 C1 F0 BE 23 01 00 B8 00 00 00 00 00 00 00   .ÃÁ..#..........

000000c0  00 00 00 02 02 01 03 00 00 00 00 00 00 00 00 00   ................
000000d0  00 00 00 00 00 00 00 00 00 00 00 DB 00 00 00 00   ...........Û....
000000e0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000000f0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000100  80 08 04 0D 0B 01 48 00 01 75 54 02 82 04 04 01   ......H..uT.....
00000110  8F 04 06 01 01 16 0E F0 90 00 00 14 0F 14 2D 80   ..............-.
00000120  15 08 15 08 00 00 00 00 00 00 00 00 00 00 00 00   ................
00000130  00 00 00 00 00 00 00 00 00 00 00 00 00 00 02 A3   ...............£
00000140  CE 00 00 00 00 00 00 00 01 4D 33 20 37 38 53 36   Î........M3 78S6
00000150  34 35 30 44 54 43 2D 43 37 35 20 43 44 34 02 25   450DTC-C75 CD4.%
00000160  39 CE 15 FF 59 41 46 33 37 30 41 FF FF FF FF FF   9Î..YAF370A.....
00000170  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
00000180  53 57 33 32 34 30 30 30 31 31 00 30 32 31 00 20   SW32400011.021.

00000190  20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ...............
000001a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000001b0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000001c0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000001d0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000001e0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000001f0  00 00 00 00 00 00 00 00 00 00 00 00 00 4A 21 73   .............J!s
>>>
```

❶ Example 4–7 shows a hex dump to DPR location 2b00, ending at block 0.

Syntax

hd [-{byte|word|long|quad}] [-{sb|eb} <n>] <file>[:<offset>].

Arguments	
<file>[:**<offset>**]	Specifies the file (byte stream) to be displayed.
Options	
-byte	Print out data in byte sizes
-word	Print out data by word
-long	Print out data by longword
-quad	Print out data by quadword
-sb <n>	Start block
-eb <n>	End block

4.10 info

The info command displays registers and data structures. You can enter the command by itself or followed by a number (0 – 8). If you do not specify a number, a list of selections is displayed and you are prompted to enter a selection.

The following commands are available:

info 0 Displays the SRM memory descriptors as described in the *Alpha System Reference Manual*.

info 1 Displays the page table entries (PTE) used by the console and operating system to map virtual to physical memory. Valid data is displayed only after a boot operation.

info 2 Dumps the Galaxy Configuration Tree (GCT) FRU table. Galaxy is a software architecture that allows multiple instances of *OpenVMS* to execute cooperatively on a single computer.

info 3 Dumps the contents of the system control status registers (CSRs) for the C-chip, D-chip, and P-chips.

info 4 Displays the per CPU impure area in abbreviated form. The console uses this scratch area to save processor context.

info 5 Displays the per CPU impure area in full form.

info 6 Displays the per CPU machine check logout area.

info 7 Displays the contents of the Console Data Log.

info 8 Clears all event frames in the Console Data Log.

For information about the data displayed by the **info** commands, see the following documents:

- For **info 0**, **info 1**, and **info 4**, see the *Alpha System Reference Manual*.
- For **info 2,** see the Galaxy Console and Alpha Systems V5.0 FRU Configuration Tree Specification.
- For **info 3,** see the Titan 21274 Chipset Functional Specification.
- For **info 6** and **info 7**, see the AlphaServer DS15 Platform Fault Management Specification.

Example 4–8 info 0

```
>>>info
```

```
         0. HWRPB MEMDSC
         1. Console PTE
         2. GCT/FRU 5
         3. Dump System CSRs
         4. IMPURE area (abbreviated)
         5. IMPURE area (full)
         6. LOGOUT area
         7. Dump Error Log
         8. Clear Error Log
Enter selection: 0

HWRPB: 2000    MEMDSC:25c0   Cluster count: 3

Cluster: 0, Usage: Console
START_PFN: 00000000  PFN_COUNT: 0000015b PFN_TESTED: 00000000
     347 pages from 0000000000000000 to 00000000002b5fff

Cluster: 1, Usage: System
START_PFN: 0000015b  PFN_COUNT: 0003fe9c PFN_TESTED: 0003fe9c
BITMAP_VA: 0000000000000000 BITMAP_PA: 00000000001be020
  261788 good pages from 00000000002b6000 to 000000007ffedfff

Cluster: 2, Usage: Console
START_PFN: 0003fff7  PFN_COUNT: 00000009 PFN_TESTED: 00000000
       9 pages from 000000007ffee000 to 000000007fffffff
>>>
```

Example 4–9 shows an **info 1** display. This output is available only after a boot operation.

Example 4–9 info 1

```
>>> info 1
pte 000000003FFA8000 0000000100001101  va 0000000010000000  pa 0000000000002000
pte 000000003FFA8008 0000000200001101  va 0000000010002000  pa 0000000000004000
pte 000000003FFA8010 0000000300001101  va 0000000010004000  pa 0000000000006000
pte 000000003FFA8018 0000000400001101  va 0000000010006000  pa 0000000000008000
pte 000000003FFA8020 0000000500001101  va 0000000010008000  pa 000000000000A000
pte 000000003FFA8028 0000000600001101  va 000000001000A000  pa 000000000000C000
pte 000000003FFA8030 0000000700001101  va 000000001000C000  pa 000000000000E000
pte 000000003FFA8038 0000000800001101  va 000000001000E000  pa 0000000000010000
pte 000000003FFA8040 0000000900001101  va 0000000010010000  pa 0000000000012000
pte 000000003FFA8048 0000000A00001101  va 0000000010012000  pa 0000000000014000
pte 000000003FFA8050 0000000B00001101  va 0000000010014000  pa 0000000000016000
pte 000000003FFA8058 0000000C00001101  va 0000000010016000  pa 0000000000018000
pte 000000003FFA8060 0000000D00001101  va 0000000010018000  pa 000000000001A000
pte 000000003FFA8068 0000000E00001101  va 000000001001A000  pa 000000000001C000
pte 000000003FFA8070 0000000F00001101  va 000000001001C000  pa 000000000001E000
pte 000000003FFA8078 0000001000001101  va 000000001001E000  pa 0000000000020000
pte 000000003FFA8080 0000001100001101  va 0000000010020000  pa 0000000000022000
pte 000000003FFA8088 0000001200001101  va 0000000010022000  pa 0000000000024000
pte 000000003FFA8090 0000001300001101  va 0000000010024000  pa 0000000000026000
pte 000000003FFA8098 0000001400001101  va 0000000010026000  pa 0000000000028000
pte 000000003FFA80A0 0000001500001101  va 0000000010028000  pa 000000000002A000
pte 000000003FFA80A8 0000001600001101  va 000000001002A000  pa 000000000002C000
pte 000000003FFA80B0 0000001700001101  va 000000001002C000  pa 000000000002E000
pte 000000003FFA80B8 0000001800001101  va 000000001002E000  pa 0000000000030000
pte 000000003FFA80C0 0000001900001101  va 0000000010030000  pa 0000000000032000
pte 000000003FFA80C8 0000001A00001101  va 0000000010032000  pa 0000000000034000
pte 000000003FFA80D0 0000001B00001101  va 0000000010034000  pa 0000000000036000
pte 000000003FFA80D8 0000001C00001101  va 0000000010036000  pa 0000000000038000
pte 000000003FFA80E0 0000001D00001101  va 0000000010038000  pa 000000000003A000
pte 000000003FFA80E8 0000001E00001101  va 000000001003A000  pa 000000000003C000
        .
        .
        .
```

Example 4–10 shows an **info 2** display. This command is the SRM's view of the configuration tree that the RCM displays.

Example 4–10 info 2

```
>>>info 2

GCT_ROOT_NODE

GCT_NODE:                  1f0000
type                       1
subtype                    0
hd_extension               0
size                       10000
rev_major                  6
rev_minor                  0
id                         0000000000000000
node_flags                 0
saved_owner                0
affinity                   0
parent                     0
child                      2c0
fw_usage                   0
Root->lock                 ffffffff
Root->transient_level      1a
Root->current_level        1a
Root->console_req          200000
Root->min_alloc            100000
Root->min_align            100000
Root->base_alloc           2000000
Root->base_align           2000000
Root->max_phys_addr        7fffffff
Root->mem_size             80000000
Root->platform_type        140400000022
Root->platform_name        0000000000000280
Root->primary_instance     0
Root->first_free           3610
Root->high_limit           fcc0
Root->lookaside            0

Root->available            bef0
Root->max_partition        1
Root->partitions           0000000000000180
Root->communities          00000000000001c0
Root->bindings             0000000000000200
Root->max_plat_partition   1
Root->max_desc             1
Root->galaxy_id            1f0128
Root->root_flags           3

dump depth view ? (Y/<N>)

dump each node ? (Y/<N>)
```

```
show flags? ( Y/<N>)

Dump a Node - Enter Handle (hex) ?

show fw_usage flags? ( Y/<N>)

>>>
```

Example 4–11 shows an **info 3** display.

Example 4–11 info 3

```
CCHIP   CSRs:             801a0000000
CSC                   7053888009192A2C : 0000
MTR                   00002F860F001225 : 0040
MISC                  00000012000000E0 : 0080
AAR0                  0000000000007009 : 0100
AAR1                  0000000000000000 : 0140
AAR2                  0000000040007009 : 0180
AAR3                  0000000000000000 : 01c0
DIM0                  D084000010003010 : 0200
DIM1                  0000000000000000 : 0240
DIM2                  0000000000000000 : 0600
DIM3                  0000000000000000 : 0640
DIR0                  0000000000000000 : 0280
DIR1                  0000000000000000 : 02c0
DIR2                  0000000000000000 : 0680
DIR3                  0000000000000000 : 06c0
DRIR                  0000000000000000 : 0300
TTR                   000000000000077F : 0580
TDR                   F7FFF7FFF7FFF7FF : 05c0

DCHIP   CSRs:             801b0000000
DSC                   3C3C3C3C3C3C3C3C : 0800
DSC2                  3C3C3C3C3C3C3C3C : 08c0
STR                   2A2A2A2A2A2A2A2A : 0840
DREV                  1111111111111111 : 0880

PCHIP 0 CSRs:             80180000000
GWSBA0                0000000000800000 : 0000
GWSBA1                0000000080000001 : 0040
GWSBA2                0000000000000000 : 0080
GWSBA3                0000000000000002 : 00c0
GWSM0                 0000000000700000 : 0100
GWSM1                 000000003FF00000 : 0140
GWSM2                 000000003FF00000 : 0180
GWSM3                 00000000FFF00000 : 01c0
GTBA0                 0000000000000000 : 0200
GTBA1                 0000000000000000 : 0240
GTBA2                 0000000004700000 : 0280
GTBA3                 0000000004800000 : 02c0
GPCTL                 00000004C10000C2 : 0300
GPLAT                 000000000000FF00 : 0340
SERROR                0000000000000000 : 0400
SERREN                000000000000000E : 0440
```

```
GPERROR              0000000000000000 : 0500
GPERREN              00000000000007F6 : 0540
SCTL                 0000000002831411 : 0700
AWSBA0               0000000000800000 : 1000
AWSBA1               0000000080000001 : 1040
AWSBA2               0000000000000000 : 1080
AWSBA3               0000000000000002 : 10c0
AWSM0                0000000000700000 : 1100
AWSM1                000000003FF00000 : 1140
AWSM2                000000003FF00000 : 1180
AWSM3                00000000FFF00000 : 11c0
ATBA0                0000000000000000 : 1200
ATBA1                0000000000000000 : 1240
ATBA2                0000000004C00000 : 1280
ATBA3                0000000005000000 : 12c0
APCTL                00000004C00200C2 : 1300
APLAT                000000000000FF00 : 1340
AGPERROR             0020000000000000 : 1400
AGPERREN             0000000000000000 : 1440
APERROR              00200000003B8000 : 1500
APERREN              00000000000007F6 : 1540
>>>
```

Example 4–12 shows an **info 4** display.

Example 4–12 info 4

```
>>>info 4
                              cpu00
per_cpu impure area           00004200
cns$flag                      0000001 : 0000
cns$flag+4                    00000000 : 0004
cns$hlt                       00000000 : 0008
cns$hlt+4                     00000000 : 000c
cns$mchkflag                  00000228 : 0210
cns$mchkflag+4                00000000 : 0214
cns$fpcr                      00000000 : 0318
cns$fpcr+4                    8ff00000 : 031c
cns$va                        001bc000 : 0320
cns$va+4                      00000000 : 0324
cns$va_ctl                    00000000 : 0328
cns$va_ctl+4                  00000000 : 032c
cns$exc_addr                  00602000 : 0330
cns$exc_addr+4                00000000 : 0334
cns$ier_cm                    00000000 : 0338
cns$ier_cm+4                  00000020 : 033c
cns$sirr                      00000000 : 0340
cns$sirr+4                    00000000 : 0344
cns$isum                      00000000 : 0348
cns$isum+4                    00000000 : 034c
cns$exc_sum                   00001fc0 : 0350
cns$exc_sum+4                 00000000 : 0354
cns$pal_base                  00008000 : 0358
cns$pal_base+4                00000000 : 035c
```

```
cns$i_ctl                     21300386 : 0360
cns$i_ctl+4                   00000000 : 0364
cns$pctr_ctl                  00000000 : 0368
cns$pctr_ctl+4                00000000 : 036c
cns$process_context           00000004 : 0370
cns$process_context+          00000000 : 0374
cns$i_stat          c         0000000 : 0378
cns$i_stat+4                  00000143 : 037c
cns$dtb_alt_mode              00000000 : 0380
cns$dtb_alt_mode+4            00000000 : 0384
cns$mm_stat                   000000b0 : 0388
cns$mm_stat+4                 00000000 : 038c
cns$m_ctl                     00000020 : 0390
cns$m_ctl+4                   00000000 : 0394
cns$dc_ctl                    000000c3 : 0398
cns$dc_ctl+4                  00000000 : 039c
cns$dc_stat                   00000000 : 03a0
cns$dc_stat+4                 00000000 : 03a4
cns$write_many                00000000 : 03a8
cns$write_many+4              00000000 : 03ac
cns$virbnd                    00000000 : 03b0
cns$virbnd+4                  00000000 : 03b4
cns$sysptbr                   00000000 : 03b8
cns$sysptbr+4                 00000000 : 03bc
cns$report_lam                00000000 : 03c0
cns$report_lam+4              00000000 : 03c4
cns$report_cstat0             00000000 : 03c8
cns$report_cstat0+4           00000000 : 03cc
cns$crd_count                 00000000 : 03d0
cns$crd_count+4               00000000 : 03d4
cns$m_fix                     00000000 : 03d8
cns$m_fix+4                   00000000 : 03dc
>>>
```

Example 4–13 shows an **info 5** display.

Example 4–13 info 5

```
>>>info 5
                      cpu00
per_cpu impure area  00004200
cns$flag             00000001 : 0000
cns$flag+4           00000000 : 0004
cns$hlt              00000000 : 0008
cns$hlt+4            00000000 : 000c
cns$gpr[0]           a124fa00 : 0010
cns$gpr[0]+4         fffffe04 : 0014
cns$gpr[1]           00000000 : 0018
cns$gpr[1]+4         00000000 : 001c
cns$gpr[2]           00000000 : 0020
cns$gpr[2]+4         00000000 : 0024
cns$gpr[3]           00000000 : 0028
cns$gpr[3]+4         00000000 : 002c
cns$gpr[4]           00000001 : 0030
cns$gpr[4]+4         00000000 : 0034
cns$gpr[5]           00000000 : 0038
cns$gpr[5]+4         00000000 : 003c
cns$gpr[6]           00000000 : 0040
cns$gpr[6]+4         00000000 : 0044
cns$gpr[7]           00000001 : 0048
cns$gpr[7]+4         00000000 : 004c
cns$gpr[8]           00000000 : 0050
cns$gpr[8]+4         00000000 : 0054
cns$gpr[9]           00a2c2b0 : 0058
cns$gpr[9]+4         fffffc00 : 005c
cns$gpr[10]          00000000 : 0060
cns$gpr[10]+4        00000000 : 0064
cns$gpr[11]          00a9eee0 : 0068
cns$gpr[11]+4        fffffc00 : 006c
cns$gpr[12]          00000000 : 0070
cns$gpr[12]+4        00000000 : 0074
cns$gpr[13]          55419000 : 0078
cns$gpr[13]+4        fffffc00 : 007c
cns$gpr[14]          7fe0e700 : 0080
cns$gpr[14]+4        fffffc00 : 0084
cns$gpr[15]          00000003 : 0088
cns$gpr[15]+4        00000000 : 008c
cns$gpr[16]          00007fff : 0090
cns$gpr[16]+4        00000000 : 0094
cns$gpr[17]          00a2c2b0 : 0098
cns$gpr[17]+4        fffffc00 : 009c
cns$gpr[18]          009978c8 : 00a0
cns$gpr[18]+4        fffffc00 : 00a4
cns$gpr[19]          00561780 : 00a8
cns$gpr[19]+4        fffffe04 : 00ac
cns$gpr[20]          00549280 : 00b0
cns$gpr[20]+4        fffffe04 : 00b4
cns$gpr[21]          5a52df80 : 00b8
cns$gpr[21]+4        fffffc00 : 00bc
cns$gpr[22]          00549280 : 00c0
cns$gpr[22]+4        fffffe04 : 00c4
cns$gpr[23]          00000000 : 00c8
cns$gpr[23]+4        00000000 : 00cc
cns$gpr[24]          00549280 : 00d0
cns$gpr[24]+4        fffffe04 : 00d4
```

```
cns$gpr[25]            24d0b9ca : 00d8
cns$gpr[25]+4          00000000 : 00dc
cns$gpr[26]            00781048 : 00e0
cns$gpr[26]+4          fffffc00 : 00e4
cns$gpr[27]            0096d1b0 : 00e8
cns$gpr[27]+4          fffffc00 : 00ec
cns$gpr[28]            00782550 : 00f0
cns$gpr[28]+4          fffffc00 : 00f4
cns$gpr[29]            009b6330 : 00f8
cns$gpr[29]+4          fffffc00 : 00fc
cns$gpr[30]            a124f7b0 : 0100
cns$gpr[30]+4          fffffe04 : 0104
cns$gpr[31]            00000000 : 0108
cns$gpr[31]+4          00000000 : 010c
cns$fpr[0]             00000000 : 0110
cns$fpr[0]+4           89000000 : 0114
cns$fpr[1]             9999999a : 0118
cns$fpr[1]+4           40ada999 : 011c
cns$fpr[2]             00000000 : 0120
cns$fpr[2]+4           00000000 : 0124
cns$fpr[3]             00000000 : 0128
cns$fpr[3]+4           00000000 : 012c
cns$fpr[4]             00000000 : 0130
cns$fpr[4]+4           00000000 : 0134
cns$fpr[5]             00000000 : 0138
cns$fpr[5]+4           00000000 : 013c
cns$fpr[6]             00000000 : 0140
cns$fpr[6]+4           00000000 : 0144
cns$fpr[7]             00000000 : 0148
cns$fpr[7]+4           00000000 : 014c
cns$fpr[8]             00000000 : 0150
cns$fpr[8]+4           00000000 : 0154
cns$fpr[9]             00000000 : 0158
cns$fpr[9]+4           00000000 : 015c
cns$fpr[10]            00000000 : 0160
cns$fpr[10]+4          00000000 : 0164
cns$fpr[11]            00000000 : 0168
cns$fpr[11]+4          00000000 : 016c
cns$fpr[12]            9999999a : 0170
cns$fpr[12]+4          3fb99999 : 0174
cns$fpr[13]            00000000 : 0178
cns$fpr[13]+4          00000000 : 017c
cns$fpr[14]            00002008 : 0180
cns$fpr[14]+4          00000000 : 0184
cns$fpr[15]            00000000 : 0188
cns$fpr[15]+4          00000000 : 018c
cns$fpr[16]            00000000 : 0190
cns$fpr[16]+4          00000000 : 0194
cns$fpr[17]            9999999a : 0198
cns$fpr[17]+4          40ada999 : 019c
cns$fpr[18]            00000000 : 01a0
cns$fpr[18]+4          40a00000 : 01a4
cns$fpr[19]            00000000 : 01a8
cns$fpr[19]+4          00000000 : 01ac
cns$fpr[20]            00000000 : 01b0
cns$fpr[20]+4          00000000 : 01b4
cns$fpr[21]            00000000 : 01b8
cns$fpr[21]+4          00000000 : 01bc
cns$fpr[22]            00000000 : 01c0
cns$fpr[22]+4          00000000 : 01c4
cns$fpr[23]            00000000 : 01c8
cns$fpr[23]+4          00000000 : 01cc
cns$fpr[24]            00000000 : 01d0
```

```
cns$fpr[24]+4        00000000 : 01d4
cns$fpr[25]          00000000 : 01d8
cns$fpr[25]+4        00000000 : 01dc
cns$fpr[26]          00000000 : 01e0
cns$fpr[26]+4        00000000 : 01e4
cns$fpr[27]          00000000 : 01e8
cns$fpr[27]+4        00000000 : 01ec
cns$fpr[28]          00000000 : 01f0
cns$fpr[28]+4        00000000 : 01f4
cns$fpr[29]          00000000 : 01f8
cns$fpr[29]+4        00000000 : 01fc
cns$fpr[30]          00000000 : 0200
cns$fpr[30]+4        00000000 : 0204
cns$fpr[31]          00000000 : 0208
cns$fpr[31]+4        00000000 : 020c
cns$mchkflag         00000228 : 0210
cns$mchkflag+4       00000000 : 0214
cns$pt               00004200 : 0218
cns$pt+4             00000000 : 021c
cns$whami            00000000 : 0220
cns$whami+4          00000000 : 0224
cns$scc              00000000 : 0228
cns$scc+4            00000000 : 022c
cns$prbr             00000000 : 0230
cns$prbr+4           00000000 : 0234
cns$ptbr             55418000 : 0238
cns$ptbr+4           00000000 : 023c
cns$trap             00000000 : 0240
cns$trap+4           00000000 : 0244
cns$halt_code        00000005 : 0248
cns$halt_code+4      00000000 : 024c
cns$ksp              a124f730 : 0250
cns$ksp+4            fffffe04 : 0254
cns$scbb             00000000 : 0258
cns$scbb+4           00000000 : 025c
cns$pcbb             1264fa00 : 0260
cns$pcbb+4           00000000 : 0264
cns$vptb             00000000 : 0268
cns$vptb+4           fffffe00 : 026c
cns$shadow4          00004200 : 02d8
cns$shadow4+4        00000000 : 02dc
cns$shadow5          00000000 : 02e0
cns$shadow5+4        00005b00 : 02e4
cns$shadow6          04f57f11 : 02e8
cns$shadow6+4        00000000 : 02ec
cns$shadow7          0001fb84 : 02f0
cns$shadow7+4        00000000 : 02f4
cns$shadow20         00000005 : 02f8
cns$shadow20+4       00000000 : 02fc
cns$p_temp           00007000 : 0300
cns$p_temp+4         00000000 : 0304
cns$p_misc           00000004 : 0308
cns$p_misc+4         00000000 : 030c
cns$shadow23         007683f0 : 0310
cns$shadow23+4       fffffc00 : 0314
cns$fpcr             00000000 : 0318
cns$fpcr+4           89000000 : 031c
cns$va               00000008 : 0320
cns$va+4             00000000 : 0324
cns$va_ctl           00000000 : 0328
cns$va_ctl+4         fffffe00 : 032c
cns$exc_addr         007683f0 : 0330
cns$exc_addr+4       fffffc00 : 0334
```

```
cns$ier_cm           e0000000 : 0338
cns$ier_cm+4         0000006a : 033c
cns$sirr             00000000 : 0340
cns$sirr+4           00000000 : 0344
cns$isum             00000000 : 0348
cns$isum+4           00000000 : 034c
cns$exc_sum          00000000 : 0350
cns$exc_sum+4        00000000 : 0354
cns$pal_base         00018000 : 0358
cns$pal_base+4       00000000 : 035c
cns$i_ctl            21300396 : 0360
cns$i_ctl+4          fffffe00 : 0364
cns$pctr_ctl         00000000 : 0368
cns$pctr_ctl+4       00000000 : 036c
cns$process_context  00000000 : 0370
cns$process_context+ 00005b00 : 0374
cns$i_stat           80000000 : 0378
cns$i_stat+4         00000147 : 037c
cns$dtb_alt_mode     00000000 : 0380
cns$dtb_alt_mode+4   00000000 : 0384
cns$mm_stat          00000290 : 0388
cns$mm_stat+4        00000000 : 038c
cns$m_ctl            00000024 : 0390
cns$m_ctl+4          00000000 : 0394
cns$dc_ctl           000000c3 : 0398
cns$dc_ctl+4         00000000 : 039c
cns$dc_stat          00000000 : 03a0
cns$dc_stat+4        00000000 : 03a4
cns$write_many       00000000 : 03a8
cns$write_many+4     00000000 : 03ac
cns$virbnd           00000000 : 03b0
cns$virbnd+4         00000000 : 03b4
cns$sysptbr          00000000 : 03b8
cns$sysptbr+4        00000000 : 03bc
cns$report_lam       00000000 : 03c0
cns$report_lam+4     00000000 : 03c4
cns$report_cstat0    00000000 : 03c8
cns$report_cstat0+4  00000000 : 03cc
cns$crd_count        00000000 : 03d0
cns$crd_count+4      00000000 : 03d4
cns$m_fix            00000000 : 03d8
cns$m_fix+4          00000000 : 03dc
>>>
```

Example 4–14 show an **info 6** display.

Example 4–14 info 6

```
>>> info 6
                          cpu00
per_cpu logout area       00006000
mchk_crd__flag_frame      00000000 : 0000
mchk_crd__flag_frame+4    00000000 : 0004
mchk_crd__offsets         00000000 : 0008
mchk_crd__offsets+4       00000000 : 000c
mchk_crd__mchk_code       00000000 : 0010
mchk_crd__mchk_code+4     00000000 : 0014
```

```
mchk_crd__i_stat          00000000 : 0018
mchk_crd__i_stat+4        00000000 : 001c
mchk_crd__dc_stat         00000000 : 0020
mchk_crd__dc_stat+4       00000000 : 0024
mchk_crd__c_addr          00000000 : 0028
mchk_crd__c_addr+4        00000000 : 002c
mchk_crd__dc1_syndrome    00000000 : 0030
mchk_crd__dc1_syndrome+4  00000000 : 0034
mchk_crd__dc0_syndrome    00000000 : 0038
mchk_crd__dc0_syndrome+4  00000000 : 003c
mchk_crd__c_stat          00000000 : 0040
mchk_crd__c_stat+4        00000000 : 0044
mchk_crd__c_sts           00000000 : 0048
mchk_crd__c_sts+4         00000000 : 004c
mchk_crd__mm_stat         00000000 : 0050
mchk_crd__mm_stat+4       00000000 : 0054
mchk_crd__os_flags        00000000 : 0058
mchk_crd__os_flags+4      00000000 : 005c
mchk_crd__cchip_dirx      00000000 : 0060
mchk_crd__cchip_dirx+4    00000000 : 0064
mchk_crd__cchip_misc      00000000 : 0068
mchk_crd__cchip_misc+4    00000000 : 006c
mchk_crd__pachip0_serror  00000000 : 0070
mchk_crd__pachip0_serror+ 00000000 : 0074
mchk_crd__pachip0_aperror 00000000 : 0080
mchk_crd__pachip0_aperror 00000000 : 0084
mchk_crd__pachip0_gperror 00000000 : 0078
mchk_crd__pachip0_gperror 00000000 : 007c
mchk_crd__pachip0_agperro 00000000 : 0088
mchk_crd__pachip0_agperro 00000000 : 008c
mchk_crd__pachip1_serror  00000000 : 0090
mchk_crd__pachip1_serror+ 00000000 : 0094
mchk_crd__pachip1_aperror 00000000 : 00a0
mchk_crd__pachip1_aperror 00000000 : 00a4
mchk_crd__pachip1_gperror 00000000 : 0098
mchk_crd__pachip1_gperror 00000000 : 009c
mchk_crd__pachip1_agperro 00000000 : 00a8
mchk_crd__pachip1_agperro 00000000 : 00ac
mchk__flag_frame          000000f8 : 00b0
mchk__flag_frame+4        00000000 : 00b4
mchk__offsets             00000018 : 00b8
mchk__offsets+4           000000a0 : 00bc
mchk__mchk_code           00000202 : 00c0
mchk__mchk_code+4         00000001 : 00c4
mchk__i_stat              00000000 : 00c8
mchk__i_stat+4            00000000 : 00cc
mchk__dc_stat             00000000 : 00d0
mchk__dc_stat+4           00000000 : 00d4
mchk__c_addr              00000000 : 00d8
mchk__c_addr+4            00000000 : 00dc
mchk__dc1_syndrome        00000000 : 00e0
mchk__dc1_syndrome+4      00000000 : 00e4
mchk__dc0_syndrome        00000000 : 00e8
mchk__dc0_syndrome+4      00000000 : 00ec
mchk__c_stat              00000000 : 00f0
mchk__c_stat+4            00000000 : 00f4
mchk__c_sts               00000000 : 00f8
mchk__c_sts+4             00000000 : 00fc
mchk__mm_stat             00000000 : 0100
mchk__mm_stat+4           00000000 : 0104
mchk__exc_addr            004c3050 : 0108
mchk__exc_addr+4          fffffc00 : 010c
mchk__ier_cm              e0000000 : 0110
```

```
mchk__ier_cm+4             0000006e : 0114
mchk__isum                 00000000 : 0118
mchk__isum+4               00000002 : 011c
mchk__reserved_0           00000000 : 0120
mchk__reserved_0+4         00000000 : 0124
mchk__pal_base             00018000 : 0128
mchk__pal_base+4           00000000 : 012c
mchk__i_ctl                21300396 : 0130
mchk__i_ctl+4              fffffe00 : 0134
mchk__process_context      00000004 : 0138
mchk__process_context+4    00004380 : 013c
mchk__reserved_1           00000000 : 0140
mchk__reserved_1+4         00000000 : 0144
mchk__reserved_2           00000000 : 0148
mchk__reserved_2+4         00000000 : 014c
mchk__os_flags             00000004 : 0150
mchk__os_flags+4           00000000 : 0154
mchk__cchip_dirx           00000000 : 0158
mchk__cchip_dirx+4         40000000 : 015c
mchk__cchip_misc           000000e0 : 0160
mchk__cchip_misc+4         00000012 : 0164
mchk__pachip0_serror       56780002 : 0168
mchk__pachip0_serror+4     00000034 : 016c
mchk__pachip0_aperror      00000000 : 0178
mchk__pachip0_aperror+4    00000000 : 017c
mchk__pachip0_gperror      00000000 : 0170
mchk__pachip0_gperror+4    00000000 : 0174
mchk__pachip0_agperror     00000000 : 0180
mchk__pachip0_agperror+4   00000000 : 0184
mchk__pachip1_serror       00000000 : 0188
mchk__pachip1_serror+4     00000000 : 018c
mchk__pachip1_aperror      00000000 : 0198
mchk__pachip1_aperror+4    00000000 : 019c
mchk__pachip1_gperror      00000000 : 0190
mchk__pachip1_gperror+4    00000000 : 0194
mchk__pachip1_agperror     00000000 : 01a0
mchk__pachip1_agperror+4   00000000 : 01a4
>>>
```

Example 4–15 shows as **info 7** display.

Example 4–15 info 7

```
>>> info 7
Number of Errors Saved = 3
Error 1
 0000 : 0001000400050018       Console Uncorrectable Error Frame Header
 0008 : 0000300a190f1324        OCT 25 15:19:36
 0010 : 0000000300000170

 0000 : 00010001000c0108    Processor Machine Check Frame
 0008 : 0000000000000000        CPU ID
 0010 : 00000000000000f8        Frame Flag/Size
 0018 : 000000a000000018        Frame Offsets
 0020 : 0000000100000098        Frame Revision/Code
 0028 : 0000000020000000        I_STAT
 0030 : 0000000000000000        DC_STAT
 0038 : 0000000000040000        C_ADDR
 0040 : 0000000000000000        DC1_SYNDROME
 0048 : 0000000000000000        DC0_SYNDROME
```

```
 0050 : 0000000000000000        C_STAT
 0058 : 000000000000000d        C_STS
 0060 : 00000000000002d1        MM_STAT
 0068 : 00000000001caf00        EXC_ADDR
 0070 : 0000002280000000        IER_CM
 0078 : 0000000200000000        ISUM
 0080 : 0000000000000000        RESERVED
 0088 : 0000000000008000        PAL_BASE
 0090 : 0000000016304386        I_CTL
 0098 : 0000000000000004        PROCESS_CONTEXT
 00a0 : 0000000000000000        Reserved
 00a8 : 0000000000000000        Reserved
 00b0 : 0000000000000004        OS Flags
 00b8 : 0000000000000000        Cchip DIRx
 00c0 : 0000000000000000        Cchip MISC
 00c8 : 0000000000000000        PChip 0 SERROR
 00d0 : 0000000000000000        PChip 0 GPERROR
 00d8 : 0000000000000000        PChip 0 APERROR
 00e0 : 0000000000000000        PChip 0 AGPERROR
 00e8 : 0000000000000000        PChip 1 SERROR
 00f0 : 0000000000000000        PChip 1 GPERROR
 00f8 : 0000000000000000        PChip 1 APERROR
 0100 : 0000000000000000        PChip 1 AGPERROR

0000 : 00010008000c0110    Titan Pchip0 Extended Frame
 0008 : 0000000002831411        SCTL
 0010 : 000000000000000e        SERREN
 0018 : 00000004c00000c2        APCTL
 0020 : 00000000000007f6        APERREN
 0028 : 0000000000000000        AGPERREN
 0030 : 0000000000000000        ASPRST
 0038 : 0000000000000000        AWSBA0
 0040 : 0000000080000001        AWSBA1
 0048 : 00000000c0000003        AWSBA2
 0050 : 0000000000000002        AWSBA3
 0058 : 0000000000000000        AWSM0
 0060 : 000000003ff00000        AWSM1
 0068 : 000000003ff00000        AWSM2
 0070 : 0000000000000000        AWSM3
 0078 : 0000000000000000        ATBA0
 0080 : 0000000000000000        ATBA1
 0088 : 0000000003100000        ATBA2
 0090 : 0000000000000000        ATBA3
 0098 : 00000004c10000c2        GPCTL
 00a0 : 00000000000007f6        GPERREN
 00a8 : 0000000000000000        GSPRST
 00b0 : 0000000000800003        GWSBA0
 00b8 : 0000000080000001        GWSBA1
 00c0 : 00000000c0000003        GWSBA2
 00c8 : 0000000000000002        GWSBA3
 00d0 : 0000000000700000        GWSM0
 00d8 : 000000003ff00000        GWSM1
 00e0 : 000000003ff00000        GWSM2
 00e8 : 0000000000000000        GWSM3
 00f0 : 00000000002cc000        GTBA0
 00f8 : 0000000000000000        GTBA1
 0100 : 0000000002f00000        GTBA2
 0108 : 0000000000000000        GTBA3

Error 3

0000 : 0001000200050018    System Event Frame Header
 0008 : 00003308060e3a34        AUG  6 14:58:52
```

```
 0010 : 0000000100000080

 0000 : 00010003000c0080    System Event Frame
 0008 : 0000000000000000        CPU ID
 0010 : 0000000000000070        Frame Flag/Size
 0018 : 0000001800000018        Frame Offsets
 0020 : 0000000100000206        Frame Revision/Code
 0028 : 0000000000000000        OS Flags
 0030 : 0000000000000000        Cchip DIRx

 0038 : 0000000000000009        TIG Info
 0040 : 0000000000000000        Reserved
 0048 : 0000000000000000        RMC Override
 0050 : 0000000000000000        Power Info
 0058 : 0000000000000000        RMC Info
 0060 : 0000000000000000        Temp Info
 0068 : 0000000000000400        Fan Info
 0070 : 0000010001000000        Fatal Summary
 0078 : 0000000000000000        Reserved
```

Example 4–16 shows an **info 8**.

Example 4–16 info 8

```
>>>info 8
Number of Errors Saved = 1, Errors Being Cleared
>>>
```

4.11 kill and kill_diags

The kill and kill_diags commands terminate diagnostics that are currently executing.

Example 4–17 kill and kill_diags

```
>>>memexer 3
>>>show_status
 ID       Program      Device    Pass  Hard/Soft Bytes Written  Bytes Read
-------- ------------ --------- ------ --------- ------------- -------------
00000001         idle system         0    0    0             0             0
000003b4      memtest memory         1    0    0     155189248     155189248
000003b9      memtest memory         1    0    0     150994944     150994944
000003ed      memtest memory         1    0    0     150994944     150994944
000003f4      memtest memory         1    0    0             0             0
>>>kill_diags
>>>
```

The **kill** command terminates a specified process. The **kill_diags** command terminates all diagnostics.

Syntax

kill_diags

kill [PID. . .]

Arguments

[PID. . .] The process ID of the diagnostic to terminate. Use the **show_status** command to determine the process ID.

4.12 memexer

The memexer command runs a specified number of memory exercisers in the background. Nothing is displayed unless an error occurs. Each exerciser tests all available memory in twice the backup cache size blocks for each pass.

The following example shows no errors.

Example 4–18 memexer

```
>>>memexer 3
>>>show_status
 ID          Program    Device     Pass Hard/Soft Bytes Written  Bytes Read
-------- ------------ --------- ------ --------- ------------- -------------
00000001         idle system         0    0    0             0             0
000003b4      memtest memory         1    0    0     155189248     155189248
000003b9      memtest memory         1    0    0     150994944     150994944
000003ed      memtest memory         1    0    0     150994944     150994944
000003f4      memtest memory         1    0    0             0             0
```

The following example shows a memory compare error indicating bad DIMMs. In most cases, the failing bank and DIMM position are specified in the error message.

```
>>> memexer 3
*** Hard Error - Error #41 - Memory compare error

Diagnostic Name    ID          Device Pass  Test  Hard/Soft   03-Sept-2003
memtest            00000193    brd0   114   1      0          12:00:01
Expected value:    25c07
Received value     35c07
Failing addr:      a11848

*** ERROR – DIMM 3 Failed ***

>>> kill_diags
>>>
```

Use the **show_status** command to display the progress of the tests. Use the **kill** or **kill_diags** command to terminate the test.

Syntax

memexer [number]

Arguments

[number]	Number of memory exercisers to start. The default is 1. The number of exercisers, as well as the length of time for testing, depends on the context of the testing.

4.13 memtest

The memtest command exercises a specified section of memory. Typically memtest is run from the built-in console script. Advanced users may want to use the specific options described here.

Example 4–19 memtest

```
>>> sh mem                                      ❶


  Array        Size       Base Address    Intlv Mode
---------  ----------  ----------------  ----------
    0        1024Mb    0000000000000000    2-Way
    2        1024Mb    0000000040000000    2-Way

     2048 MB of System Memory
>>>

>>>memtest -sa 400000 -l 2000000 -p 10&
>>>
                       ❷      ❸      ❹
>>> memtest -sa 400000 -l 2000000 -p 10&
*** Hard Error - Error #43 - Memory compare error

Diagnostic Name  ID             Device  Pass  Test  Hard/Soft   8-Sept-2003
memtest          00000118       brd0     1     1     1    0     12:00:01
Expected value:                fffffffe
Received value:                ffffffff
Failing addr:                    400004
*** Error - DIMM 3 Failed ***           ❺
```

❶ Use the **show memory** command or an **info 0** command to see where memory is located.

❷ Starting address

❸ Length of the section to test in bytes

❹ Passcount. In this example, the test will run for 10 passes.

❺ The test detected a failure on DIMM 3.

Use the **show_status** command to display the progress of the test. Use the **kill** or **kill_diags** command to terminate the test.

Memtest provides a graycode memory test. The test writes to memory and then reads the previously written value for comparison. The section of memory that is tested has its data destroyed. The **-z** option allows testing outside of the main memory pool. Use caution because this option can overwrite the console.

Memtest may be run on any specified address. If the **-z** option is not included (default), the address is verified and allocated from the firmware's memory zone. If the **-z** qualifier is included, the test is started without verification of the starting address.

When a starting address is specified, the memory is allocated beginning at the starting address -32 bytes for the length specified. The extra 32 bytes that are allocated are reserved for the allocation header information. Therefore, if a starting address of 0xa00000 and a length of 0x100000 is requested, the area from 0x9fffe0 through 0xb00000 is reserved. This may be confusing if you try to begin two **memtest** processes simultaneously with one beginning at 0xa00000 for a length of 0x100000 and the other at 0xb00000 for a length of 0x100000. The second **memtest** process will send a message that it is “Unable to allocate memory of length 100000 at starting address b00000.” Instead, the second process should use the starting address of 0xb00020.

NOTE: *If **memtest** is used to test large sections of memory, testing may take a while to complete. If you issue a Ctrl/C or **kill PID** in the middle of testing, **memtest** may not abort right away. For speed reasons, a check for a Ctrl/C or **kill** is done outside of any test loops. If this is not satisfactory, you can run concurrent **memtest** processes in the background with shorter lengths within the target range.*

Memtest Test 1 — Graycode Test

Memtest Test 1 uses a graycode algorithm to test a specified section of memory. The graycode algorithm used is: data = (x>>1)^x, where x is an incrementing value.

Three passes are made of the memory under test.

- The first pass writes alternating graycode inverse graycode to each four longwords. This causes many data bits to toggle between each 16-byte write.

  For example graycode patterns for a 32 byte block would be:

  Graycode(0) 00000000 Graycode(1) 00000001 Graycode(2) 00000003 Graycode(3) 00000002 Inverse Graycode(4) FFFFFFF9 Inverse Graycode(5) FFFFFFF8 Inverse Graycode(6) FFFFFFFA Inverse Graycode(7) FFFFFFFB
- The second pass reads each location, verifies the data, and writes the inverse of the data, one longword at a time. This causes all data bits to be written as a one and zero.
- The third pass reads and verifies each location.

You can specify the **-f** (fast) option so that the explicit data verify sections of the second and third loops are not performed. This does not catch address shorts but stresses memory with a higher throughput. The ECC/EDC logic can be used to detect failures.

Syntax

memtest ([-sa <start_address>] [-ea <end_address>] [-l <length>]
[-bs <block_size>] [-i <address_inc>] [-p <pass_count>]
[-d <data_pattern>] [-rs <random_seed>] [-ba <block_address>]
[-t <test_mask>] [-se <soft_error_threshold>]
[-g <group_name>] [-rb] [-f] [-m] [-z] [-h] [-mb])

Options

-sa	Start address. Default is first free space in memzone.
-ea	End address. Default is start address plus length size.
-l	Length of section to test in bytes, default is the zone size with the -**rb** option and the block_size for all other tests. **-l** has precedence over **-ea**.
-bs	Block (packet) size in bytes in hex, default 8192 bytes. This is used only for the random block test. For all other tests the block size equals the length.
-i	Specifies the address increment value in longwords. This value is used to increment the address through the memory to be tested. The default is 1 (longword). This is only implemented for the graycode test. An address increment of 2 tests every other longword. This option is useful for multiple CPUs testing the same physical memory.
-p	Passcount If 0 then run indefinitely or until Ctrl/C is issued. Default = 1
-t	Test mask. Default = run all tests in selected group.
-g	Group name
-se	Soft error threshold
-f	Fast. If **-f** is included in the command line, the data compare is omitted. Detects only ECC/EDC errors.

Options

-m Timer. Prints out the run time of the pass. Default = off .

-z Tests the specified memory address without allocation. Bypasses all checking but allows testing in addresses outside of the main memory heap. Also allows unaligned input.

CAUTION: *This flag can overwrite the console. If the system hangs, press the halt/reset button (if configured for reset).*

-d Used only for march test (2). Uses this pattern as test pattern. Default = 5's

-h Allocates test memory from the firmware heap.

-rs Used only for random test (3). Uses this data as the random seed to vary random data patterns generated. Default = 0.

-rb Randomly allocates and tests all of the specified memory address range. Allocations are done of block_size.

-mb Memory barrier flag. Used only in the **-f** graycode test. When set an mb is done after every memory access. This guarantees serial access to memory.

-ba Used only for block test (4). Uses the data stored at this address to write to each block.

4.14 net

The net command performs maintenance operations on a specified Ethernet port. Net -ic initializes the MOP counters for the specified Ethernet port, and net -s displays the current status of the port, including the contents of the MOP counters.

Example 4–20 net -ic and net -s

```
>>>net -ic eia0
>>>net -s eia0
i82558 Statistics:
 TX:
 Good Frames: 22, Max Collisions: 0, Late Collisions: 0
 Underruns: 0, Lost CRS: 0, Deferred: 0
 Single Coll.: 0, Multiple Coll.: 0, Total Coll.: 0
 RX:
 Good Frames: 15, CRC errors: 0, Align errors: 0
 Rx not ready: 0, Overrun: 0, Coll Detect: 0, Short Frames: 0

 RU Restarts: 0, CU Restarts: 0, CU Timeouts: 0

MOP BLOCK:
 Network list size: 0

MOP COUNTERS:
Time since zeroed (Secs): 12

TX:
 Bytes: 0 Frames: 0
 Deferred: 0 One collision: 0 Multi collisions: 0
TX Failures:
 Excessive collisions: 0 Carrier check: 0 Short circuit: 0
 Open circuit: 0 Long frame: 0 Remote defer: 0
 Collision detect: 0

RX:
 Bytes: 0 Frames: 0
 Multicast bytes: 0 Multicast frames: 0
RX Failures:
 Block check: 0 Framing error: 0 Long frame: 0
 Unknown destination: 0 Data overrun: 0 No system buffer: 0
 No user buffers: 0
```

Syntax

net [-ic]

net [-s]

Arguments

<port_name>	Specifies the Ethernet port on which to operate, either eg*0, ei*0, or ew*0.

4.15 nettest

The nettest command tests the network ports using MOP loopback. Typically nettest is run from the built-in console script. Advanced users may want to use the specific options and environment variables described here.

Example 4–21 nettest

```
>>> nettest ei*                      ❶
>>> nettest -mode in ew*             ❷
>>> nettest -mode ex -w 10  e*       ❸
```

❶ Internal loopback test on port ei*0

❷ Internal loopback test on ports ewa0/ewb0

❸ External loopback test on all Ethernet ports; wait 10 seconds between tests

Nettest performs a network test. It can test the ei* or ew* ports in internal loopback, external loopback, or live network loopback mode.

Nettest contains the basic options to run MOP loopback tests. Many environment variables can be set from the console to customize **nettest** before **nettest** is started. The environment variables, a brief description, and their default values are listed in the syntax table in this section. Each variable name is preceded by e*a0_ or e*b0_ to specify the desired port.

You can change other network driver characteristics by modifying the port mode. See the **-mode** option.

Use the **show_status** display to determine the process ID when terminating an individual diagnostic test. Use the **kill** or **kill_diags** command to terminate tests.

Syntax

nettest ([-f <file>] [-mode <port_mode>] [-p <pass_count>] [-sv <mop_version>] [-to <loop_time>] [-w <wait_time>] [<port>])

Arguments

<port>	Specifies the Ethernet port on which to run the test.

Options

-f <file>	Specifies the file containing the list of network station addresses to loop messages to. The default file name is lp_nodes_e*a0 for port e*a0. The default file name is lp_nodes_e*b0 for port e*b0. The files by default have their own station address.
-mode <port_mode>	Specifies the mode to set the port adapter. The default is ex (external loopback). Allowed values are: df : default, use environment variable values ex : external loopback in : internal loopback nm : normal mode nf : normal filter pr : promiscuous mc : multicast ip : internal loopback and promiscuous fc : force collisions nofc : do not force collisions nc : do not change mode
-p <pass_count>	Specifies the number of times to run the test. If 0, then run until terminated by a **kill** or **kill_diags** command The default is 1.

NOTE: *This is the number of passes for the diagnostic. Each pass will send the number of loop messages as set by the environment variable,* **ega*_loop_count, eia*_loop_count,** *or* **ewa*_loop_count**.

-sv <mop_version>	Specifies which MOP version protocol to use. If 3, then MOP V3 (DECNET Phase IV) packet format is used. If 4, then MOP V4 (DECNET Phase V IEEE 802.3) format is used.

-to <loop_time>	Specifies the time in seconds allowed for the loop messages to be returned. The default is 2 seconds.
-w <wait_time>	Specifies the time in seconds to wait between passes of the test. The default is 0 (no delay). The network device can be very CPU intensive. This option will allow other processes to run.

Environment Variables

e*a*_loop_count	Specifies the number (hex) of loop requests to send. The default is 0x3E8 loop packets.
e*a*_loop_inc	Specifies the number (hex) of bytes the message size is increased on successive messages. The default is 0xA bytes.
e*a*_loop_patt	Specifies the data pattern (hex) for the loop messages. The following are legitimate values.
	0 : all zeros
	1 : all ones
	2 : all fives
	3 : all 0xAs
	4 : incrementing data
	5 : decrementing data
	ffffffff : all patterns
loop_size	Specifies the size (hex) of the loop message. The default packet size is 0x2E.

4.16 set sys_serial_num

The set sys_serial_num command sets the system serial number. This command is used by Manufacturing for establishing the system serial number, which is then propagated to all FRU devices that have EEPROMs. The sys_serial_num environment variable can be read by the operating system.

IMPORTANT: *The system serial number must be set correctly. System Event Analyzer will not work with an incorrect serial number.*

Example 4–22 set sys_serial_num

```
>>> set sys_serial_num NI900100022
```

When the system motherboard is replaced, you must use the **set sys_serial_num** command to restore the master setting.

Syntax

set sys_serial_num *value*

Value is the system serial number, which is on a sticker on the back of the system enclosure.

4.17 show error

The show error command reports errors logged to the FRU EEPROMs.

Example 4–23 show error

```
>>> show error
❶
HMB           TDD - Type: 15 Test: 15  SubTest: 15  Error: 15              ❷
001f8408   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F   ................
HMB           SDD - Type: 14 LastLog: 0   Overwrite: 0                     ❸
001f8408   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F   ................
001f8418   0F 0F 0F 0F 0F 0F 0F 0F 0F 00 00 00 00 00 00 00   ................
001f8428   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8438   00 00 00 00 00 00 00 00 00 00 00 FF 00 00 00 00   ................
001f8448   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8458   00 00 00 00 00 00 00 00                           ........
HMB           Bad checksum   0 to 64   EXP:dc   RCV:dd                     ❹
001f8408   80 08 00 01 53 00 01 00 00 00 00 00 00 00 00 00   ....S...........
001f8418   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8428   FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8438   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 DD   ...............Y
HMB           Bad checksum  64 to 126  EXP:e1  RCV:0f
001f8408   4A FF FF FF FF FF FF FF 02 35 34 2D 31 32 33 34   J........54-1234
001f8418   35 2D 30 31 2E 41 30 30 31 20 20 00 00 09 44 91   5-01.A001  ...D.
001f8428   34 51 15 41 41 41 41 41 41 41 41 41 41 41 41 41   4Q.AAAAAAAAAAAAA
001f8438   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F   ................
HMB           Bad checksum 128 to 254  EXP:0c  RCV:0d
001f8408   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F   ................
001f8418   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F   ................
001f8428   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F   ................
001f8438   0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 00 00   ................
001f8408   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8418   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8428   FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
001f8438   00 00 00 00 00 00 00 00 00 00 00 00 00 4A 21 0D   .............J!.
HMB           SYS_SERIAL_NUM Mismatch                                       ❺
>>>
```

The output of the **show error** command is based on information logged to the serial control bus EEPROMs on the system FRUs. Both the operating system and the ROM-based diagnostics log errors to the EEPROMs. This functionality allows you to generate an error log from the console environment. No errors are displayed for fans and the power supply because these components do not have an EEPROM.

Syntax

show error

All FRUs with errors are displayed. If no errors are logged, nothing is displayed and you are returned to the SRM console prompt.

Example 4–23 shows TDD, SDD, checksum, and sys_serial_num mismatch errors logged to the EEPROM on the system motherboard (HMB). Table 4–2 shows a reference to these errors. The bit masks correspond to the bit masks that would be displayed in the E field of the **show fru** command.

❶ FRU to which errors are logged; in this example the system motherboard, HMB.

❷ A TDD error has been logged. TDDs (test-directed diagnostics) test specific functions sequentially. Typically, nothing else is running during the test. TDDs are performed in SROM or XSROM or early in the console power-up flow.

❸ An SDD error has been logged. SDDs (symptom-directed diagnostics) are generic diagnostic exercisers that try to cause random behavior and look for failures or "symptoms." All SDDs are logged by the System Event Analyzer.

❹ Three checksum errors have been logged.

❺ There was a mismatch between the serial number on the system motherboard and the system serial number. This could occur if a motherboard from a system with a different serial number was swapped into this system.

4.18 show fru

The show fru command displays the physical configuration of FRUs. Use show fru -e to display FRUs with errors. FRUs with EEPROMs are normally set in ICT at manufacturing.

Example 4–24 show fru

```
>>> build hmb 54-30558-01.b1 ay94412345
>>> show fru
❶               ❷    ❸                    ❹            ❺          ❻

FRUname          E   Part#               Serial#    Model/Other Alias/Misc
HMB              00 54-30558-01.B1      SW32400011       -

HMB.DIMM0.J14    00 20-01EBA09          SW32400011       234          ce

HMB.DIMM1.J12    00 20-01EBA09          SW32400011       234          ce

HMB.DIMM2.J15    00 20-01EBA09          SW32400011       234          ce

HMB.DIMM3.J13    00 20-01EBA09          SW32400011       234          ce

HMB.PCIRSR       00 54-30560-01.A1      SW31200018       -

HMB.PCIRSR.PCI   00                                     3D Labs OX
PWR0             00 30-10005-0-         -               PS
SYS FAN          00 12-10010-01                         FAN         J3
PCI FAN          00 12-49806-04                         FAN         J1
CPU FAN          00 12-56450-01                         FAN         J32
DISK FAN         00 12-45971-04                         FAN         J31
>>>
```

❶ FRUname — The FRU name recognized by the SRM console. The name also indicates the location of that FRU in the physical hierarchy.

HMB = system motherboard; CPU = CPUs; DIMMn = DIMMs; CPB = PCI; PCI = PCI option; SBM = SCSI backplane; PWR = power supply; FAN = fans; JIO= I/O connector module (junk I/O).

❷ E — Error field. Indicates whether the FRU has any errors logged against it. FRUs without errors show 00 (hex). FRUs with errors have a non-zero value that represents a bit mask of possible errors. See Table 4–2.

❸ Part # — The part number of the FRU in ASCII, either an HP part number or a vendor part number.

❹	Serial #	The serial number. For HP FRUs, the serial number has the form XXYWWNNNNN. XX = manufacturing location code YWW = year and week NNNNN = sequence number. For vendor FRUs, the 4-byte sequence number is displayed in hex.
❺	Model/Other	Optional data. For HP FRUs, the HP part alias number (if one exists). For vendor FRUs, the year and week of manufacture.
❻	Alias/Misc	Miscellaneous information about the FRUs. For HP FRUs, a model name, number, or the common name for the entry in the Part # field. For vendor FRUs, the manufacturer's name.

The following table lists bit assignments for failures that could potentially be listed in the E (error) field of the **show fru** command. Because the E field is only two characters wide, bits are "'or'ed" together if the device has multiple errors. For example, the E field for a FRU with both TDD (02) and SDD (04) errors would be 06:

010 | 100 = 110 (6)

Table 4–2 Show Error Message Translation

Bit Mask (E Field)	Text Message	Meaning and Action
01	<*fruname*> Hardware Failure	Module failure. FRUs that are known to be connected but are unreadable are considered hardware failures. An example is power supplies.
02	<*fruname*> TDD - Type:0 Test: 0 SubTest: Error: 0	Serious error. Run the System Event Analyzer (SEA), if necessary, to determine what action to take. If you cannot run SEA, replace the module.
04	<*fruname*> SDD - Type:0 LastLog: 0 Overwrite: 0	Serious error. SEA has written a FRU callout into the SDD area and DPR global area. Follow the instructions given by SEA.
08	<*fruname*> EEPROM Unreadable	Reserved.
10	<*fruname*> Bad checksum 0 to 64 EXP:01 RCV:02	Informational. Use the **clear_error** command to clear the error unless TDD or SDD is also set.
20	<*fruname*> Bad checksum 64 to 126 EXP:01 RCV:02	Informational. Use the **clear_error** command to clear the error unless TDD or SDD is also set.

Bit Mask (E Field)	Text Message	Meaning and Action
40	<*fruname*> Bad checksum 128 to 254 EXP:01 RCV:02	Informational. Use the **clear_error** command to clear the error unless TDD or SDD is also set.
80	<*fruname*> SYS_SERIAL_NUM Mismatch	Informational. Use the **clear_error** command to clear the error unless TDD or SDD is also set.

4.19 show_status

The show_status command displays the progress of diagnostics. The command reports one line of information per executing diagnostic. Many of the diagnostics run in the background and provide information only if an error occurs.

Example 4–25 show _status

```
>>> show_status

❶             ❷            ❸            ❹     ❺          ❻              ❼

 ID           Program     Device       Pass  Hard/Soft Bytes Written  Bytes Read
-------- ------------ ----------- ------ --------- ------------- -----------
00000001         idle system          0    0    0             0           0
00002147      memtest memory          1    0    0     742391808   742391808
0000214c      memtest memory          1    0    0     742391808   742391808
00002151      memtest memory          2    0    0     729808896   729808896
0000218b      memtest memory          1    0    0             0           0
0000218c      memtest memory          2    0    0     734003200   734003200
000021cf     exer_kid dka0.0.0.8.0    0    0    0             0      483328
000021d0     exer_kid dka100.1.0.8    0    0    0             0      483328
000021df     exer_kid dqa0.0.0.13.    0    0    0             0      482304
00002211     exer_kid tta1            4    0    0          4252        4252

0000227b      nettest eia0.0.0.9.0   38    0    0         53504       53504
000022d4      nettest eib0.0.0.10.   37    0    0         52096       52096
>>>
```

❶ Process ID

❷ The SRM diagnostic for the particular device

❸ The device under test

❹ Number of diagnostic passes that have been completed

❺ Error count (hard and soft). Soft errors are not usually fatal; hard errors halt the system or prevent completion of the diagnostics.

❻ Bytes successfully written by the diagnostic.

❼ Bytes successfully read by the diagnostic.

The following command string is useful for periodically displaying diagnostic status information for diagnostics running in the background:

```
>>> while true;show_status;sleep n;done
```

Where *n* is the number of seconds between **show_status** displays.

Syntax

show_status

4.20 sys_exer

The sys_exer command exercises the devices displayed with the show config command. Tests are run concurrently and in the background. Nothing is displayed after the initial test startup messages unless an error occurs.

Example 4–26 sys_exer

```
>>>sys_exer
Default zone extended at the expense of memzone.
Use INIT before booting
Exercising the Memory
Exercising the DK* Disks (read-only)
Exercising the DQ* Disks (read-only)
Testing the VGA (Alphanumeric Mode only)
Exercising the EI* Network

Type "show_status" to display testing progress
Type "cat el" to redisplay recent errors
Type "init" in order to boot the operating system

>>>show_status
 ID       Program      Device       Pass  Hard/Soft Bytes Written  Bytes Read
-------- ---------- ------------ ------ --------- ------------- -------------
00000001       idle system           0    0    0             0             0
000031bb    memtest memory           1    0    0     339738624     339738624
000031c0    memtest memory           1    0    0     335544320     335544320
000031c5    memtest memory           1    0    0     335544320     335544320
000031f9    memtest memory           1    0    0     327155712     327155712

00003200    memtest memory           1    0    0             0             0
00003243   exer_kid dka0.0.0.8.0     0    0    0             0        156672
00003244   exer_kid dka100.1.0.8     0    0    0             0        156672
00003253   exer_kid dqa0.0.0.13.     0    0    0             0        197632
000032bc    nettest eia0.0.0.9.0    17    0    0         23936         23936
00003318    nettest eib0.0.0.10.    15    0    0         21270         21120
00003318    nettest lp_nodes_eib    15    0    0             0             6
>>>init
Initializing...


OpenVMS PALcode V1.98-6, Tru64 UNIX PALcode V1.92-7

starting console on CPU 0
```

Use the **show_status** command to display the progress of diagnostic tests.
The diagnostics started by the **sys_exer** command automatically reallocate memory resources, because these tests require additional resources. Use the **init** command to reconfigure memory before booting an operating system.

Because the **sys_exer** tests are run concurrently and indefinitely (until you stop them with the **init** command), they are useful in flushing out intermittent hardware problems.

When using the **sys_exer** command after shutting down an operating system, you must initialize the system to a quiescent state. Enter the following command at the SRM console:

```
>>> init
.
.
.
>>> sys_exer
```

By default, no write tests are performed on disk and tape drives. Media must be installed to test the drives. When the **-lb** argument is used, a loopback connector is required for the COM2 port (9-pin loopback connector, 12-27351-01).

Syntax

sys_exer [-lb] [-t]

Arguments

[-lb]	The loopback option runs console loopback tests for the COM2 serial port during the test sequence.
[-t]	Number of seconds to run. The default is run until terminated by a **kill** or **kill_diags** command.

4.21 test

The test command verifies all the devices in the system. This command can be used on all supported operating systems.

Example 4–27 test -lb

```
>>>test -lb
Testing the Memory (full)
.
No DY* Disks available for testing
No DZ* Disks available for testing
Testing the DK* Disks (read only)
No DR* Disks available for testing
No DQ* Disks available for testing
No DF* Disks available for testing
No MK* Tapes available for testing
No MU* Tapes available for testing
Testing the Serial Port 1(external loopback)
Testing the VGA (Alphanumeric Mode only)
Testing the EI* Network
>>>
```

The **test** command also does a quick test on the system speaker. A beep is emitted as the command starts to run.

The tests are run sequentially, and the status of each subsystem test is displayed to the console terminal as the tests progress. If a particular device is not available to test, a message is displayed. The test script does no destructive testing; that is, it does not write to disk drives.

Syntax

test [*argument*]

Use the **-lb** (loopback) argument for console loopback tests.

To run a complete diagnostic test using the **test** command, the system configuration must include:

- A serial loopback connected to the COM2 port (not included)
- A trial CD-ROM with files installed

The test script tests devices in the following order:

1. Memory tests (one pass)
2. Read-only tests: DK* disks, DR* disks, DQ* disks, and MK* tapes

---

NOTE: *You must install media to test disks and tape drives. Since no write tests are performed, it is safe to test disks and tapes that contain data.*

---

3. Console loopback tests if **-lb** argument is specified: COM2 serial port.
4. VGA console tests: These tests are run only if the console environment variable is set to **serial**. The VGA console test displays rows of the word *HP*.
5. Network internal loopback tests for EW*, EI*, and EG* networks.

Chapter 5
Error Logs

This chapter explains how to interpret error logs reported by the operating system. The following topics are covered:

- Error Log Analysis with System Event Analyzer
- Fault Detection and Reporting
- Machine Checks/Interrupts

5.1 Error Log Analysis with System Event Analyzer

System Event Analyzer (SEA) is a fault management diagnostic tool that is used to determine the cause of hardware failures. System Event Analyzer performs system diagnostic processing of both single and multiple error/fault events.

System Event Analyzer may or may not be installed on the customer's system with the operating system, depending on the release cycle. If SEA is installed, the System Event Analyzer Director starts automatically as part of the system start-up. SEA provides automatic background analysis. When an error event occurs, it triggers the firing of an analysis rule. The analysis engine collects and processes the information and typically generates a “problem found” report, if appropriate. The report can be automatically sent to users on a notification mailing list and, if DSNlink is installed, a call can be logged with the customer support center.

System Event Analyzer has the capability to support the *Tru64 UNIX* and *OpenVMS* operating systems on Alpha platforms.

NOTE: *Compaq Analyze was a successor tool to DECevent and typically did not support the same systems as DECevent. Compaq Analyze was renamed to System Event Analyzer in release V4.2.*

5.1.1 WEB Enterprise Service (WEBES) Director

System Event Analyzer uses the functionality contained in the WEBES Director, a process that manages all other WEBES processes and executes continuously on the machine when configured to do so. The Director manages the decomposition processing of system error events, provides required information to the analysis engine, and performs notification message routing for the system. System Event Analyzer provides the functionality for system event analysis and Bit-To-Text (BTT) translation.

System Event Analyzer includes common WEBES code. Subsequent releases of System Event Analyzer will continue to ship with the common WEBES code.

The Director is started when the system is booted. Normally you do not need to start the Director. If the Director has stopped running, restart it by following the instructions in the *System Event Analyzer User Guide*.

System Event Analyzer includes a graphical user interface (WUI) that allows the user to interact with the Director. While only one Director process executes on the machine at any time, many WUI processes can run at the same time, connected to the single Director. Refer to the System Event Analyzer installation and user manuals for the respective operating system to launch the System Event Analyzer WUI. The HP service tools Web sites available to customers are:

http://h18023.www1.hp.com/support/svctools/webes

or

http://www.compaq.com/support/svctools/webes

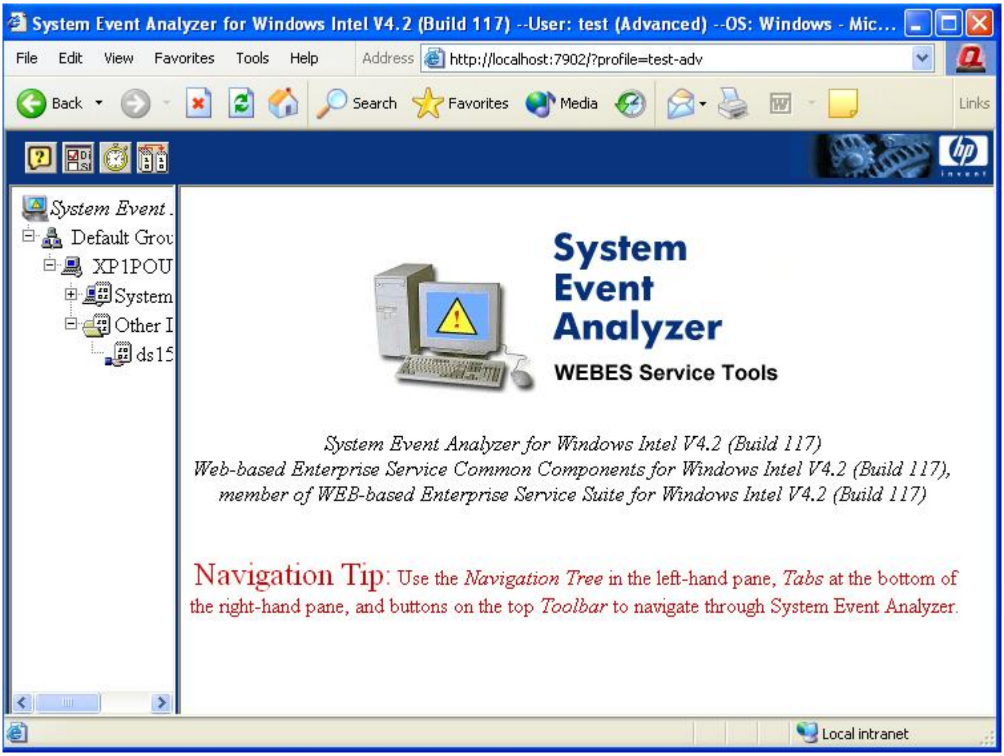
The applicable System Event Analyzer documentation includes the following:

- *System Event Analyzer User's Guide*
- *WEBES Installation Guide for Tru64 UNIX*
- *WEBES Installation Guide for OpenVMS*
- *System Event Analyzer Releases Notes*
- *WEBES Releases Notes*

5.1.2 Using System Event Analyzer

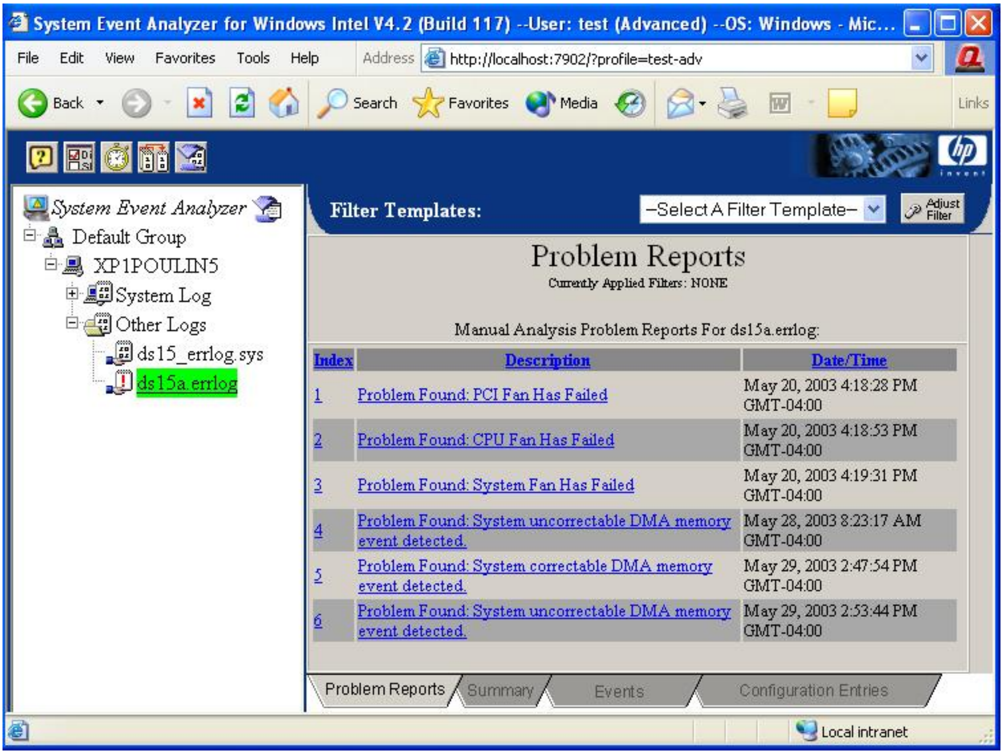
After you have logged on to System Event Analyzer the following screen appears. If an event has occurred, it is listed under "localhost" events. See Figure 5–1.

Figure 5–1 System Event Analyzer Initial Screen



In Figure 5–2, the Other Logs file is selected and the list of Problem Reports is displayed.

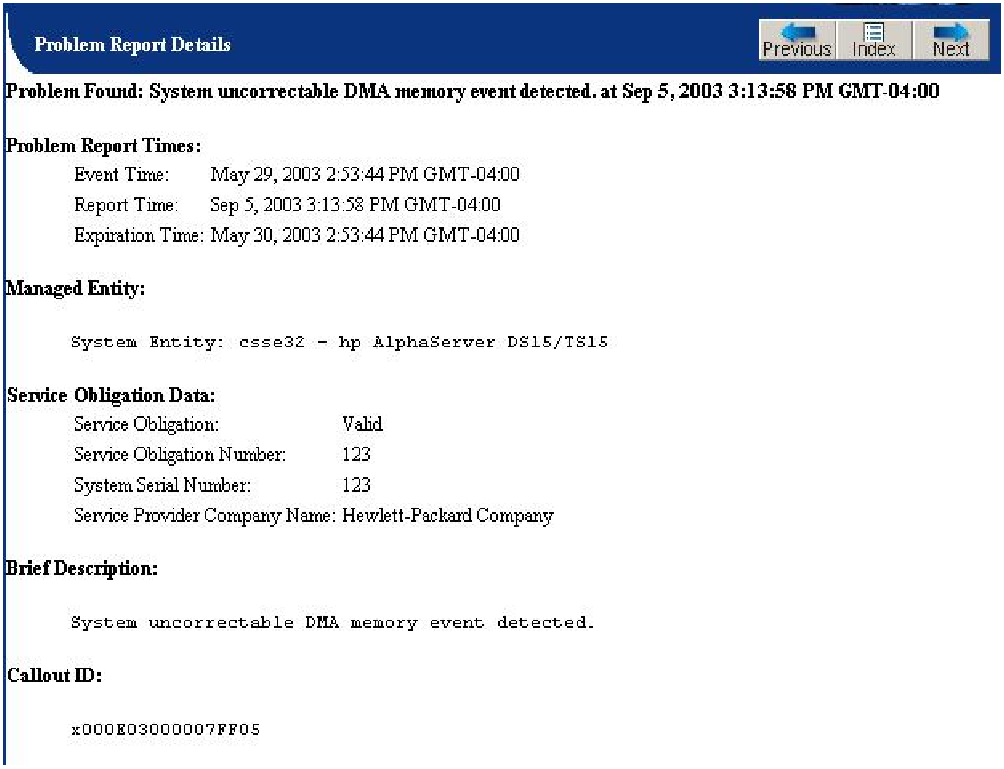
Figure 5–2 Problem Reports Screen



Full View is selected and the problem reports are listed. You may select any log listed in Other Logs to view a list of all problems found. You may also view each report by clicking on the underlined hot link under Problem Reports.

Figure 5–3 provides an example of a problem report.

Figure 5–3 System Event Analyzer Problem Report Details



Problem Report Details

Previous | Index | Next

Problem Found: System uncorrectable DMA memory event detected. at Sep 5, 2003 3:13:58 PM GMT-04:00

Problem Report Times:

Event Time:	May 29, 2003 2:53:44 PM GMT-04:00
Report Time:	Sep 5, 2003 3:13:58 PM GMT-04:00
Expiration Time:	May 30, 2003 2:53:44 PM GMT-04:00

Managed Entity:

```
System Entity: csse32 - hp AlphaServer DS15/TS15
```

Service Obligation Data:

Service Obligation:	Valid
Service Obligation Number:	123
System Serial Number:	123
Service Provider Company Name:	Hewlett-Packard Company

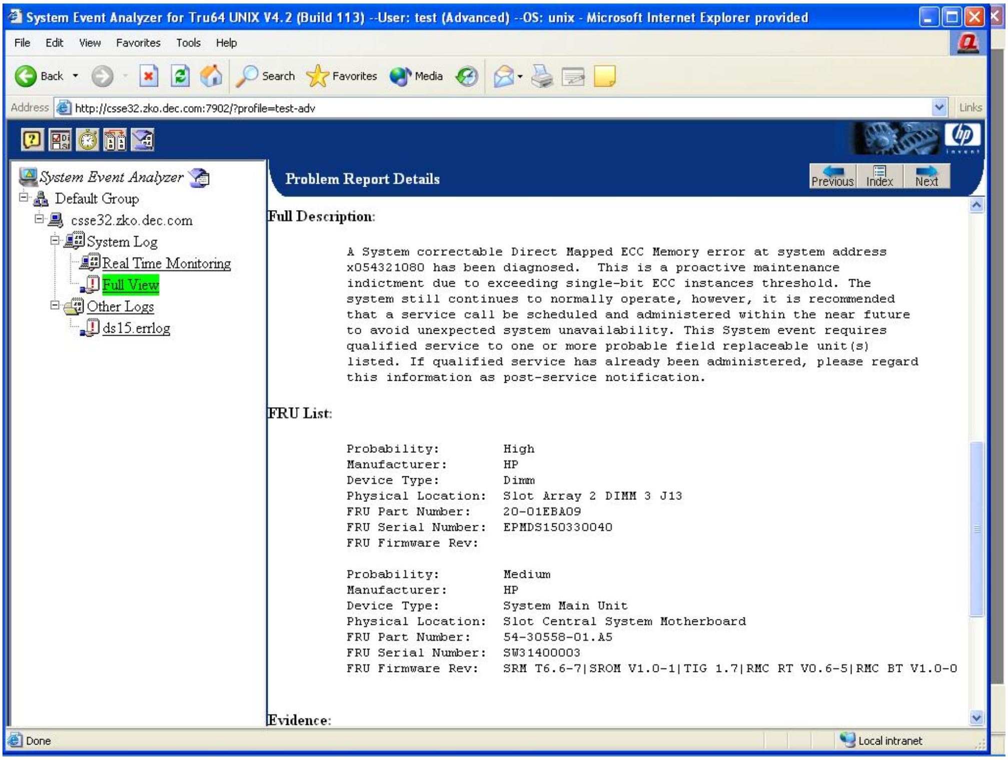
Brief Description:

```
System uncorrectable DMA memory event detected.
```

Callout ID:

```
x000E03000007FF05
```

Figure 5–3 System Event Analyzer Problem Report Details (Continued)



Managed Entity

The Managed Entity designator includes the system host name (typically a computer name for networking purposes), the type of computer system (AlphaServer DS15), and the error event identification. The error event identification uses new common event header Event_ID_Prefix and Event_ID_Count components. The Event_ID_Prefix refers to an OS specific identification for this event type. The Event_ID_Count indicates the number of this event and the event type.

Service Obligation Data

This item provides Obligation number and validity, system serial number, and company name of service provider.

Brief Description

The Brief Description designator indicates whether the error event is related to the CPU, system (PCI, storage, and so on), or environmental subsystem.

Callout ID

The Callout ID designator provides information about the analysis rule-set. Most characters within this designator are reserved for HP-specific purposes.

Full Description

The Full Description designator provides detailed error information, which can include a description of the detected fault or error condition, the specific address or data bit where this fault or error occurred, the probable FRU list, and service related information.

FRU List

The FRU List designator lists the most probable defective FRUs. This list indicates that one or more of these FRUs needs to be serviced. The information typically includes the FRU probability, manufacturer, system device type, system physical location, part number, serial number, and firmware revision level (if applicable).

5.1.3 Bit to Text

The following is an example of the Correctable System Event for ds15a.errlog. To access the data, select the Events tab for the problem report selected.

NOTE: *1. By default, SEA does not display correctable cpu or system events (event type 630/620). To display these events with the WUI, the "-adv" must be added to the logon profile, for example: test-adv*
2. When using the CLI to translate event type 630 or 620, the showall qualifier must be added to the command, for example:
"wsea x trans ds15a.errlog showall".

Figure 5–4 Correctable System Event Sample Table

```
Event:       35
Description: Correctable System Event at May 29, 2003 2:44:50 PM GMT-04:00
from csse32
File:        ds15a.errlog
============================================================================

COMMON EVENT HEADER (CEH) V2.0
Event_Leader           xFFFF FFFE
Header_Length           256
Event_Length            560
Header_Rev_Major        2
Header_Rev_Minor        0
OS_Type                 1                     -- Tru64 UNIX
Hardware_Arch           4                     -- Alpha
CEH_Vendor_ID           3,564                 -- Hewlett-Packard Company
Hdwr_Sys_Type           38                    -- Titan Corelogic
Logging_CPU             0                     -- CPU Logging this Event
CPUs_In_Active_Set      1
Major_Class             100
Minor_Class             3
Entry_Type              620                   -- Correctable System Event
DSR_Msg_Num             2,047                 -- AlphaServer DS15 1Ghz
Chip_Type               12                    -- EV68CB - 21264C
CEH_Device              37
CEH_Device_ID_0        x0000 03FF
CEH_Device_ID_1        x0000 0007
CEH_Device_ID_2        x0000 0007
Unique_ID_Count         2
Unique_ID_Prefix        63,376
Num_Strings             5

TLV Section of CEH
TLV_DSR_String         AlphaServer DS15
TLV_OS_Version         Compaq Tru64 UNIX V5.1B (Rev. 2650)
TLV_Sys_Serial_Num     EPMDS15033
TLV_Time_as_Local      May 29, 2003 2:44:50 PM GMT-04:00
TLV_Computer_Name      csse32
Entry_Type              620

Logout_Frame_CPU_Section
Frame_Size             x0000 00B0
Frame_Flags            x8000 0000

CPU_Area_Offset        x0000 0018
System_Area_Offset     x0000 0058
Mchk_Error_Code        x0000 0204           Machine Check Logout Frame Error
Code
   Value[31:0]         x204                   System Non-Fatal
Frame_Rev              x0000 0001
I_STAT                 x0000 0000 0000 0000
DC_STAT                x0000 0000 0000 0000
C_ADDR                 x0000 0000 0000 0000   Cbox Read Erred Address
Register
   Io_M[43]            x0                     System Memory Access
C_SYNDROME_1           x0000 0000 0000 0000   Odd QW Data Syndrome
   QW_Upper[7:0]       x0                     No Syndrome
C_SYNDROME_0           x0000 0000 0000 0000   Even QW Data Syndrome
   QW_Lower[7:0]       x0                     No Syndrome
C_STAT                 x0000 0000 0000 0000
```

```
   Cbox_Error[4:0]      x0
C_STS                   x0000 0000 0000 0000    Cache Block Access Status
Register
   Cblock_Status[3:0]   x0                      Status unknown
MM_STAT                 x0000 0000 0000 0000    Memory Management Status
Register

Logout_Frame_System_Section
SW_Error_Sum_Flags      x0000 0000 0000 0004
   Pchip0_PCI_Error[0]  x0
   Pchip1_PCI_Error[1]  x0
   Pchip_Mem_Error[2]   x1                      Pchip or CPU Memory  Error
Detected
   Hot_Plug_Slot[39:32]x0
Cchip_DIRx              x1000 0000 0000 0000    Cchip Device Interrupt Request
Register
   Pchip0_Cerr[60]      x1                      Pchip 0 Non-Fatal Error
Detected
Cchip_MISC              x0000 0012 0000 00E0    Cchip Miscellaneous Register
   Nxm[28]              x0                      Nxs[31:29] NOT Valid
   Nxs[31:29]           x0                      If Nxm[28] = 1 - CPU 0 Source
Device
P0_Serror               xF100 0023 4999 0004    Pchip0 System Error Register
   Corr_Ecc_Error[2]    x1                      Non-Fatal ECC Error
   Sys_Addr[46:15]      x46 9332                System Erred Address Bits
[34:3]
   Bus_Source[53:52]    x0                      GPCI Bus
   TransAction_Cmd[55:54]x0                       DMA Read
   ECC_Syndrome[63:56]  xF1                     Data Bit 30
                        xF1                     Error ECC Syndrome
P0_GPerror              x0000 0000 0000 0000    No Error Detected
   PCI_Cmd[55:52]       x0                      Interrupt Acknowledge
P0_APerror              x0020 0000 003B 4000    Pchip0 Aport Error Register
   PCI_Addr[46:14]      xED                     PCI Erred Address Bits [34:2]
   PCI_Cmd[55:52]       x2                      IO Read
P0_AGPerror             x0000 0000 0000 0000    No Error Detected
   AGP_Lost_Err[0]      x0
   AGP_Cmd[52:50]       x0                      Read
P1_Serror               x0000 0000 0000 0000    No Error Detected
   Bus_Source[53:52]    x0                      GPCI Bus
   TransAction_Cmd[55:54]x0                       DMA Read
   ECC_Syndrome[63:56]  x0                      No ECC Error
P1_GPerror              x0000 0000 0000 0000    No Error Detected
   PCI_Cmd[55:52]       x0                      Interrupt Acknowledge
P1_APerror              x0000 0000 0000 0000    No Error Detected
   PCI_Cmd[55:52]       x0                      Interrupt Acknowledge
P1_AGPerror             x0000 0000 0000 0000    No Error Detected

START OF SUBPACKETS IN THIS EVENT

ES4X Dual Port RAM Subpacket, Version 1
DPR_0                   x40                     Non - Split, Set0 - 4 Dimms
(DS15
- 2 Dimms), configured as lowest array
DPR_1                   x10                     Array 0 Dpr Location x81
   Array_0_Size[7:0]    x10                     1 Gbytes
DPR_2                   x00                     DPR Location x82 Unused
DPR_3                   x00                     DPR Location x83 Unused
   Array_1_Size[7:0]    x0                      No Good Memory in Array 1
DPR_4                   x41                     Non - Split, Set0 - 4 Dimms
(DS15
- 2 Dimms), configured as next lowest array
DPR_5                   x10                     Array 2 Dpr Location x85
```

```
   Array_2_Size[7:0]   x10                      1 Gbytes
DPR_6                  x00                      DPR Location x86 Unused
DPR_7                  x00                      DPR Location x87 Unused
   Array_3_Size[7:0]   x0                       No Good Memory in Array 3

System Memory / IO Configuration Subpacket, Version 1
AAR_0                  x0000 0000 0000 7009     Memory Array 0 Configuration
Register
   Sa0[8]              x0                       Non - Split  Array
   Asiz0[15:12]        x7                       1 Gb
   Addr0[34:24]        x0                       Array0 Base Address [34:24]
Bits
AAR_1                  x0000 0000 0000 0000     Memory  Array 1 Configuration
Register
   Sa1[8]              x0                       Non - Split  Array
   Asiz1[15:12]        x0                       Array 1 Not Used
   Addr1[34:24]        x0                       Array1 Base Address [34:24]
Bits
AAR_2                  x0000 0000 4000 7009     Memory  Array 2 Configuration
Register
   Sa2[8]              x0                       Non - Split  Array
   Asiz2[15:12]        x7                       1 Gb
   Addr2[34:24]        x40                      Array2 Base Address [34:24]
Bits
AAR_3                  x0000 0000 0000 0000     Memory  Array 3 Configuration
Register
   Sa3[8]              x0                       Non - Split  Array
   Asiz3[15:12]        x0                       Array 3 Not Used
   Addr3[34:24]        x0                       Array3 Base Address [34:24]
Bits
P0_SCTL                x0000 0000 0283 1411     Pchip0 System Control Register
   REV[7:0]            x11
   PID[8]              x0                       Pchip ID Value
   RPP[9]              x0
   ECCEN[10]           x1                       DMA ECC Enabled
   SWARB[12:11]        x2                       Round Robin
   CRQMAX[19:16]       x3
   CDQMAX[23:20]       x8
   PTPMAX[27:24]       x2
   INUM[28]            x0                       256K Max Downstream PTP/PIO
Writes to bypass PIO Read
   NEWAMU[29]          x0                       GPCI Enabled to Perform PTE
Fetch
Xactions
   PTPWAR[30]          x0                       PTP Writes Disabled During
Pending Reads
P0_GPCTL               x0000 0004 C100 00C2     Pchip 0 Gport Control Register
   FBTB[0]             x0
   THDIS[1]            x1                       TLB Anti-Thrash Disabled
   CHAINDIS[2]         x0
   TGTLAT[4:3]         x0                       Target Latency Timer = 128 PCI
Clocks
   Win_HOLE[5]         x0
   MnStr_WIN_Enable[6] x1                       Monster Window Enabled
   ARBENA[7]           x1                       Pchip 0 Internal Arbiter
Enabled
   PRIGRP[15:8]        x0                       No req_l[6:0] High Priority
   PPRI[16]            x0
   PCISPD66[17]        x0                       GPCI Frequency = 33 MHz
   CNGSTLT[21:18]      x0                       All DMA Reads Retry w/delayed
Completion
   PTPDESTEN[29:22]    x4                       Writes to Pchip0, APCI
Enabled
```

```
   DPCEN[30]           x1                       Data Parity Checking Enabled
   APCEN[31]           x1                       Address Parity Checking
Enabled
   DCR_Timer[33:32]    x0                       DCR Timer Count = 2^15
   EN_Stepping[34]     x1                       Address Stepping Enabled
P0_APCTL               x0000 0004 C002 00C2     Pchip0 Aport Control Register
   FBTB[0]             x0
   THDIS[1]            x1                       TLB Anti-Thrashing Disabled
   CHAINDIS[2]         x0
   TGLAT[4:3]          x0                       TGLAT = 128 PCI Clocks
   HOLE_Enable[5]      x0
   MWIN_Enable[6]      x1                       Monster Window Enabled
   ARBENA[7]           x1                       Internal Arbiter Enabled
   PRIGRP[15:8]        x0
   PCISPD66[17]        x1                       APCI = 66MHz
   CNGSTLT[21:18]      x0                       All DMA Reads Retry w/delayed
Completion
   PTPDESTEN[29:22]    x0                       No Legal PTPs Enabled
   DPCEN[30]           x1
   APCEN[31]           x1
   DCR_Timer[33:32]    x0                       DCRT Count = 2^15
   EN_Stepping[34]     x1                       PCI Config Address Stepping
Enabled
   AGP_Rate[53:52]     x0                       AGP Rate = 1X
   AGP_SBA_Enabled[54] x0
   AGP_Enabled[55]     x0
   AGP_Present[57]     x0                       PCI Bus Enabled
   AGP_HP_RD[60:58]    x0                       0 Cchip HP Outstanding Reads
   AGP_LP_RD[63:61]    x0                       0 Cchip LP Outstanding Reads
P1_SCTL                x0000 0000 0000 0000     Pchip1 System Control Register
   REV[7:0]            x0
   PID[8]              x0                       Pchip PID = 0
   RPP[9]              x0
   ECCEN[10]           x0                       Pchip1 ECC Disabled
   SWARB[12:11]        x0                       GPCI > APCI > AGPX
   CRQMAX[19:16]       x0
   CDQMAX[23:20]       x0
   PTPMAX[27:24]       x0
   INUM[28]            x0                       256K MAX PTP/PIO Writes
Enabled
to bypass PIO Read
   NEWAMU[29]          x0                       GPCI Enabled to Perform PTE
Fetch
Xactions
   PTPWAR[30]          x0                       PTP Writes Disabled During
Pending Reads
P1_GPCTL               x0000 0000 0000 0000     Pchip1 Gport Control Register
   FBTB[0]             x0                       PCI Fast Back-To_Back Xactions
Disabled
   THDIS[1]            x0                       TLB Anti-Thrashing Disabled
   CHAINDIS[2]         x0                       GPCI PIO Write Chaining
Disabled
   TGLAT[4:3]          x0                       Target RetryTimer = 128 PCI
Clocks
   WIN_Hole[5]         x0                       512Kb - 1Mb Window Hole
Disabled
   Mnstr_Win_Enable[6] x0                       Monster Window Disabled
   ARBENA[7]           x0                       Internal Arbiter Disabled
   PRIGRP[15:8]        x0                       No Arbitor Priority Groups
Enabled
   PCISPD66[17]        x0                       GPCI Frequency = 33 Mhz
   CNGSTLT[21:18]      x0                       Every DMA Read Retry w/Delayed
Completion Enabled
```

```
   PTPDESTEN[29:22]    x0                      All GPCI Legal PTP
Destinations
Disabled
   DPCEN[30]           x0                      Data Parity Error Detection
Disabled
   APCEN[31]           x0                      Address Parity Error Detection
Disabled
   DCRTV[33:32]        x0                      DCR Timer = 2^15 Counts
   EN_Stepping[34]     x0                      Address Stepping Disabled
P1_APCTL               x0000 0000 0000 0000    Pchip1 Aport Control Register
   FBTB[0]             x0                      PCI Fast Back-To-Back Xactions
Disabled
   THDIS[1]            x0                      TLB Anti-Thrashing Enabled
   CHAINDIS[2]         x0                      APCI PIO Write Chaining
Enabled
   TGLAT[4:3]          x0                      Target Latency Timer = 128 PCI
Clocks
   Win_Hole[5]         x0                      512Kb - 1Mb Hole Disabled
   Mnstr_Win_Enable[6] x0                      Monster Window Disabled
   ARBENA[7]           x0                      Arbiter Disabled
   PRIGRP[15:8]        x0                      No High Priority Groups
Enabled
   PPRI[16]            x0
   PCISPD66[17]        x0                      APCI Frequency = 33 Mhz
   CNGSLT[21:18]       x0                      Every DMA Read Retry w/Delayed
Completion Enabled
   PTPDESTEN[29:22]    x0                      All APCI legal PTP
Destinations
Disabled
   DPCEN[30]           x0                      Data Parity Error Detection
Disabled
   APCEN[31]           x0                      Address Command Parity Error
Detection Disabled
   DCRTV[33:32]        x0                      DCR Timer = 2^15 Counts
   EN_Stepping[34]     x0                      Address Stepping Disabled
   AGP_Rate[53:52]     x0                      AGP Rate = 1X
   AGP_SBA_EN[54]      x0                      SideBand Addressing Disabled
   AGP_EN[55]          x0                      AGP Xactions Disabled
   AGP_Present[57]     x0                      agp_present = 0
   AGP_HP_RD[60:58]    x0                      No Cchip Pending HP Reads
   AGP_LP_RD[63:61]    x0                      No Cchip Pending LP Reads
```

5.2 Fault Detection and Reporting

Table 5–1 provides a summary of the fault detection and correction components of DS15 systems.

Generally, PALcode handles exceptions/interrupts as follows:

1. The PALcode determines the cause of the exception/interrupt.
2. If possible, it corrects the problem and passes control to the operating system for error notification, reporting, and logging before returning the system to normal operation.
3. If PALcode is unable to correct the problem, it
   - Logs double error halt error frames into the flash ROM
   - Logs uncorrectable error logout frames to the DPR
   - For single error halts, logs the uncorrectable logout frame into the DPR.
4. If error/event logging is required, control is passed through the OS Privileged Architecture Library (PAL) handler. The operating system error handler logs the error condition into the binary error log. System Event Analyzer should then diagnose the error to the defective FRU.

Table 5–1 DS15 Fault Detection and Correction

Component	Fault Detection/Correction Capability
Alpha 21264C (EV68) microprocessor	Contains error checking and correction (ECC) logic for data cycles. Check bits are associated with all data entering and exiting the microprocessor. A single-bit error on any of the four longwords being read can be corrected (per cycle). A double-bit error on any of the four longwords being read can be detected (per cycle).
Backup cache (B-cache)	ECC check bits on the data store, and parity on the tag address store and tag control store.
Memory DIMMs	ECC logic protects data by detecting and correcting data cycle errors. A single-bit error on any of the four longwords can be corrected (per cycle). A double-bit error on any of the four longwords being read can be detected (per cycle).
PCI SCSI controller adapter	SCSI data parity is generated.
PCI devices	Data and address parity is checked.

5.3 Machine Checks/Interrupts

The exceptions that result from hardware system errors are called machine checks/interrupts. They occur when a system error is detected during the processing of a data request.

During the error-handling process, errors are first handled by the appropriate PALcode error routine and then by the associated operating system error handler. PALcode transfers control to the operating system through the PAL handler.

Table 5–2 lists the machine checks/interrupts that are related to error events. The designations – 630, 670, 620, 660, and 680 – indicate a system control block (SCB) offset to the fatal system error handler for *Tru64 UNIX* and *OpenVMS*.

Table 5–2 Machine Checks/Interrupts

Error Type	Error Descriptions
CPU Correctable Error (630) Generic Alpha 21264C (EV68) correctable errors.	B-cache probe hit single-bit ECC error D-cache tag parity error on issue I-cache tag or data parity error D-cache victim single-bit ECC error B-cache single-bit ECC fill error to I-stream or D-stream Memory single-bit ECC fill error to I-stream or D-stream
CPU Uncorrectable Error (670) Fatal microprocessor machine check errors that result in a system crash.	PAL detected bugcheck error Operating system detected bugcheck error EV68 detected second D-cache store EEC error EV68 detected D-cache tag parity error in pipeline 0 or 1 EV68 detected duplicate D-cache tag parity error EV68 detected double-bit ECC memory fill error EV68 detected double-bit probe hit EEC error EV68 detected B-cache tag parity error

Table 5–2 Machine Checks/Interrupts (Continued)

Error Type	Error Descriptions
System Correctable Error (620) DS15-specific correctable errors.	System detected ECC single-bit error
System Uncorrectable Error (660) A system-detected machine check that occurred as a result of an "off-chip" request to the system.	Uncorrectable ECC error Nonexistent memory reference PCI system bus error (SERR) PCI read data parity error (RDPE) PCI address/command parity error (APE) PCI no device select (NDS) PCI target abort (TA) Invalid scatter/gather page table entry (SGE) error PCI data parity error (PERR) Flash ROM write error PCI target delayed completion retry time-out (DCRTO) PCI master retry time-out (RTO 2**24) error PCI-ISA software NMI error
System Environmental Error (680) System-detected machine check caused by an overtemperature condition, fan failure, or power supply failure.	Warning Threshold 45 degrees C. Fatal/Power-down Threshold 50 degrees C. Emergency Thermal limit 75 degrees C. (see Note) Complete power supply failure Fan failure and warnings Power supply failure (redundant supply) RMC internal errors

NOTE: *The override for fan and overtemperature shutdown is set in the RMC. If override is set, the system continues operating until more severe errors occurs.*

5.3.1 Error Logging and Event Log Entry Format

The operating system error handlers generate several entry types that vary in length based on the number of registers within the entry.

Each entry consists of an operating system header, several device frames, and an end frame. Most entries have a PAL-generated logout frame, and may contain frames for CPU, memory, and I/O. Table 5–3 shows an event structure map for a system uncorrectable PCI target abort error. An *AlphaServer* DS15 has only PCHIP 0. PCHIP 1 information in error registers is always 0.

Table 5–3 Sample Error Log Event Structure Map

OFFSET(hex)	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
ech0000 ech+nnnn	NEW COMMON OS HEADER							
lfh0000 lfh+nnnn	STANDARD LOGOUT FRAME HEADER							
lfEV680000 lfEV68+nnnn	COMMON PAL EV68 SECTION (first 8 QWs Zeroed)							
lfcctt_A0[u]	SESF<63:32> = Reserved(MBZ)			<39:32>= (MBZ)	SESF<31:16> = Reserved(MBZ)		SESF<15:0>= 0001(hex)	
lfcctt_A8[u]	Cchip CPUx Device Interrupt Request Register (DIRx<62> = 1)							
lfcctt_B0[u]	Cchip Miscellaneous Register (MISC)							
lfcctt_B8[u]	Pchip0 Error Register (P0_PERROR<51>=0;<47:18>=PCI Addr;<17:16>=PCI Opn; <6>=1)							
lfcctt_C0[u]	Pchip1 Error Register (P1_PERROR<63:0> = 0)							
lfett_C8[u] lfett_138[u]	Pchip0 Extended Titan System Packet							
eelcb_140 eelcb_190 eelcb_1E0 eelcb_230	Pchip 0 PCI Slot 1 Single Device Bus Snapshot Packet Pchip 0 PCI Slot 2 Single Device Bus Snapshot Packet Pchip 0 PCI Slot 3 Single Device Bus Snapshot Packet Pchip 0 PCI Slot 4 Single Device Bus Snapshot Packet							
2D8	Termination or End Packet							

Chapter 6
System Configuration and Setup

This chapter describes how to configure and set up an *AlphaServer* DS15 system. The following topics are covered:

- System Consoles
- Displaying the Hardware Configuration
- Setting Environment Variables
- Setting Automatic Booting
- Changing the Default Boot Device
- Setting SRM Security
- Configuring Devices
- Booting Linux

6.1 System Consoles

The SRM console is located in a flash ROM on the system motherboard. From the console interface, you can set up and boot the operating system, display the system configuration, and run diagnostics. For complete information, see the *AlphaServer DS15 and AlphaStation DS15 Owner's Guide.*

SRM Console

Systems running the *Tru64 UNIX* or *OpenVMS* operating systems are configured from the SRM console, a command-line interface (CLI). From the CLI you can enter commands to configure the system, view the system configuration, boot the system, and run ROM-based diagnostics.

NOTE: *The operating systems use different algorithms for system time. If you switch between operating systems (for example, between TRU64 UNIX and OpenVMS), be sure to reset the time at the operating system level.*

Linux

The procedure for installing Linux on an Alpha system is described in the Alpha Linux installation document for your Linux distribution. The installation document can be downloaded from the following Web site:

http://www.compaq.com/alphaserver/linux

RMC CLI

The remote management console (RMC) provides a command-line interface (CLI) for controlling the system. You can use the CLI either locally or remotely (modem connection) to power the system on and off, halt or reset the system, and monitor the system environment. You can also use the **dump**, **env**, and **status** commands to help diagnose errors. See Chapter 7 for details.

6.1.1 Selecting the Display Device

The SRM console environment variable determines to which display device (VT-type terminal or VGA monitor) the console display is sent.

The console terminal that displays the SRM user interface can be either a serial terminal (VT320 or higher, or equivalent) or a VGA monitor.

The SRM **console** environment variable determines the display device.

- If **console** is set to **serial**, and a VT-type device is connected, the SRM console powers on in serial mode and sends power-up information to the VT device.
- If **console** is set to **graphics**, the SRM console expects to find a VGA card and, if so, displays power-up information on the VGA monitor after VGA initialization has been completed.

You can verify the display device with the SRM **show console** command and change the display device with the SRM **set console** command. If you change the display device setting, you must reset the system (with either the halt/reset button, if configured, the RMC **reset** command, or the SRM **init** command) to put the new setting into effect.

In the following example, the user displays the current console device (a graphics device) and then resets it to a serial device. After the system initializes, output will be displayed on the serial terminal.

```
>>> show console
console                 graphics
>>> set console serial
>>> init
.
.
.
```

6.2 Displaying the Hardware Configuration

View the system hardware configuration by entering commands from the SRM console. It is useful to view the hardware configuration to ensure that the system recognizes all devices, memory configuration, and network connections.

Use the following SRM console commands to view the system configuration. See the *Owner's Guide* for details.

show boot*	Displays the boot environment variables.
show config	Displays the logical configuration of interconnects and buses on the system and the devices found on them.
show device	Displays the bootable devices and controllers in the system.
show fru	Displays the physical configuration of FRUs (field-replaceable units).
show memory	Displays configuration of main memory.

6.3 Setting 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.

- To check the setting for a specific environment variable, enter the **show *envar*** command, where the name of the environment variable is substituted for *envar*.
- To reset an environment variable, use the **set *envar*** command, where the name of the environment variable is substituted for *envar*.

set *envar*

The **set** command sets or modifies the value of an environment variable. It can also be used to create a new environment variable if the name used is unique. 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. The syntax is:

set *envar value*

Envar	The name of the environment variable to be modified.
value	The new value of the environment variable.

New values for the following environment variables take effect only after you reset the system with either the halt/reset button, if configured, the RMC **reset** command, or the SRM **init** command.

auto_action
console
os_type
pk*0_fast
pk*0_host_id
pk*0_soft_term

show *envar*

The **show *envar*** command displays the current value (or setting) of an environment variable. The syntax is:

show *envar*

Envar	The name of the environment variable to be displayed. The wildcard * displays all environment variables.

Table 6–1 summarizes the SRM environment variables used most often on the DS15 system.

Table 6–1 SRM Environment Variables

Variable	Attributes	Description
auto_action	NV,W[1]	Action the console should take following an error halt or power failure. Defined values are: **boot**—Attempt bootstrap. **halt**—Halt, enter console I/O mode. **restart**—Attempt restart. If restart fails, try boot.
bootdef_dev	NV,W	Device or device list from which booting is to be attempted when no path is specified. Set at factory to disk with factory-installed software; otherwise **NULL**.
boot_file	NV,W	Default file name used for the primary bootstrap when no file name is specified by the **boot** command. The default value is **NULL**.
boot_osflags	NV,W	Default parameters to be passed to system software during booting if none are specified by the **boot** command. ***OpenVMS***: Additional parameters are the *root_number* and *boot flags*. The default value is **NULL**. *root_number:* Directory number of the system disk on which OpenVMS files are located. 0 (default)—[SYS0.SYSEXE] 1—[SYS1.SYSEXE] 2—[SYS2.SYSEXE] 3—[SYS3.SYSEXE]

[1] NV—Nonvolatile. The last value saved by system software or set by console commands is preserved across cold bootstraps (when the system goes through a full initialization), and long power outages.

W—Warm nonvolatile. The last value set by system software is preserved across warm bootstraps (*Tru64 UNIX* **shutdown -r** command, *OpenVMS* **REBOOT** command, or a crash and reboot; not all of the SRM initialization is run) and restarts.

Table 6–1 SRM Environment Variables (Continued)

Variable	Attributes	Description
boot_osflags (continued)	NV,W	*boot_flags:* The hexadecimal value of the bit number or numbers to set. To specify multiple boot flags, add the flag values (logical OR). 1—Bootstrap conversationally (enables you to modify SYSGEN parameters in SYSBOOT). 2—Map XDELTA to running system. 4—Stop at initial system breakpoint. 8—Perform a diagnostic bootstrap. 10—Stop at the bootstrap breakpoints. 20—Omit header from secondary bootstrap file. 80—Prompt for the name of the secondary bootstrap file. 100—Halt before secondary bootstrap. 10000—Display debug messages during booting. 20000—Display user messages during booting. ***Tru64 UNIX*:** The following parameters are used with this operating system: **a**—Autoboot. Boots /vmunix from bootdef_dev, goes to multi-user mode. Use this for a system that should come up automatically after a power failure. **s**—Stop in single-user mode. Boots /vmunix to single-user mode and stops at the # (root) prompt. **i**—Interactive boot. Requests the name of the image to boot from the specified boot device. Other flags, such as -kdebug (to enable the kernel debugger), may be entered using this option.
boot_osflags (continued)		**D**—Full dump; implies **s** as well. By default, if *Tru64 UNIX* crashes, it completes a partial memory dump. Specifying **D** forces a full dump at system crash. Common settings are **a**, autoboot, and **Da**, autoboot and create full dumps if the system crashes.

Table 6–1 SRM Environment Variables (Continued)

Variable	Attributes	Description
com1_baud	NV,W	Sets the baud rate of the COM1 (MMJ) port. The default baud rate is 9600. Baud rate values are 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200, 38400, 57600.
com2_baud	NV,W	Sets the baud rate of the COM2 port. The default baud rate is 9600. Baud rate values are 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200, 38400, 57600.
com1_flow **com2_flow**	NV,W	The **com1_flow** and **com2_flow** environment variables indicate the flow control on the serial ports. Defined values are: **none**—No data flows in or out of the serial ports. Use this setting for devices that do not recognize XON/XOFF or that would be confused by these signals. **software**—Use XON/XOFF(default). This is the setting for a standard serial terminal. **hardware**—Use modem signals CTS/RTS. Use this setting if you are connecting a modem to a serial port.
com1_mode	NV	Specifies the COM1 data flow paths so that data either flows through the RMC or bypasses it.
com1_modem	NV,W	Used to tell the operating system whether a modem is present on the COM1 port. **On**—Modem is present. **Off**—Modem is not present (default value).
console	NV	Sets the device on which power-up output is displayed. Graphics—Sets the power-up output to be displayed at a VGA monitor or device connected to the VGA module. Serial—Sets the power-up output to be displayed on the device that is connected to the COM1 port.

Continued on next page

Table 6–1 SRM Environment Variables (Continued)

Variable	Attributes	Description
eg*0_inet_init or **ei*0_inet_init** or **ew*0_inet_init**	NV	Determines whether the interface's internal Internet database is initialized from nvram or from a network server (via the bootp protocol).
eg*0_mode or **ei*0_mode** or **ew*0_mode**	NV	Sets the Ethernet controller to the default Ethernet device type. **aui**—Sets the default device to AUI. **bnc**—Sets the default device to ThinWire. **fast**—Sets the default device to fast 100BaseT. **fastfd**—Sets the default device to fast full duplex 100BaseT. **full**—Set the default device to full duplex twisted pair. **twisted-pair**—Sets the default device to 10BaseT (twisted-pair). **autonegotiate**—Automatically negotiates highest common performance with other network controller(s) supporting IEEE 802.3u auto-negotiation. If no Ethernet cable is connected in this mode, the SRM reports a failure: Error (eib0.0.10.0), No link, auto negotiation did not complete. This is applicable for ei*, ew*, and eg* device in auto negotiate.
eg*0_protocols or **ei*0_protocols** or **ew*0_protocols**	NV	Determines which network protocols are enabled for booting and other functions. **Mop**—Sets the network protocol to MOP for systems using the OpenVMS operating system. **Bootp**—Sets the network protocol to bootp for systems using the Tru64 UNIX operating system. **Bootp,mop**—When the settings are used in a list, the mop protocol is attempted first, followed by bootp.

Continued on next page

Table 6–1 SRM Environment Variables (Continued)

Variable	Attributes	Description
heap_expand	NV	Increases the amount of memory available for the SRM console's heap. Valid selections are: NONE (default) 64KB 128KB 256KB 512KB 1MB 2MB 3MB 4MB
kbd_hardware type	NV	Sets the keyboard hardware type as either PCXAL or LK411 and enables the system to interpret the terminal keyboard layout correctly.
kzpsa_host_id	W	Specifies the default value for the KZPSA host SCSI bus node ID.
language	NV	Specifies the console keyboard layout. The default is English (American).
memory_test	NV	Specifies the extent to which memory is tested prior to a boot on *Tru64 UNIX*. The options are: **Full**—Full memory test will be run. Required for *OpenVMS*. **Partial**—First 256 MB of memory will be tested. **None**—Only first 32 MB will be tested.

Table 6–1 SRM Environment Variables (Continued)

Variable	Attributes	Description
os_type	NV	Sets the default operating system. **vms** or **unix**—Sets system to boot the SRM firmware.
password	NV	Sets a console password. Required for placing the SRM into secure mode.
pci_parity	NV	Disable or enable parity checking on the PCI bus. **On**—PCI parity enabled (default value) **Off**—PCI parity disabled Some PCI devices do not implement PCI parity checking, and some have a parity-generating scheme in which the parity is sometimes incorrect or is not fully compliant with the PCI specification. In such cases, the device functions properly so long as parity is not checked.
pk*0_fast	NV	Enables fast SCSI devices on a SCSI controller to perform in standard or fast mode. **0**—Sets the default speed for devices on the controller to standard SCSI. If a controller is set to standard SCSI mode, both standard and fast SCSI devices will perform in standard mode. **1**—Sets the default speed for devices on the controller to fast SCSI mode. Devices on a controller that connects to both standard and Fast SCSI devices will automatically perform at the appropriate rate for the device, either fast or standard mode.

Continued on next page

Table 6–1 SRM Environment Variables (Continued)

Variable	Attribute	Description
pk*0_host_id	NV	Sets the controller host bus node ID to a value between 0 and 7. 0 to 7—Assigns bus node ID for specified host adapter.
pk*0_soft_term	NV	Enables or disables SCSI terminators for optional SCSI controllers. This environment variable applies to systems using the Qlogic SCSI controller, though it does not affect the onboard controller. The Qlogic SCSI controller implements the 16-bit wide SCSI bus. The Qlogic module has two terminators, one for the 8 low bits and one for the high 8 bits. There are five possible values: **off**—Turns off both low 8 bits and high 8 bits. **Low**—Turns on low 8 bits and turns off high 8 bits. **High**—Turns on high 8 bits and turns off low 8 bits. **On**—Turns on both low 8 bits and high 8 bits.
sys_serial_num	NV	Sets the system serial number, which is then propagated to all FRUs that have EEPROMs. The serial number can be read by the operating system.
tt_allow_login	NV	Enables or disables login to the SRM console firmware on alternative console ports. **0**—Disables login on alternative console ports. **1**—Enables login on alternative console ports (default setting). If the console output device is set to **serial**, **set tt_allow_login 1** allows you to log in on the primary COM1 port, or alternate COM2 port, or the VGA monitor. If the console output device is set to **graphics**, **set tt_allow_login 1** allows you to log in through either the COM1 or COM2 console port.

6.4 Setting Automatic Booting

***Tru64 UNIX* and *OpenVMS* systems are factory set to halt in the SRM console. You can change these defaults, if desired.**

Systems can boot automatically (if set to autoboot) from the default boot device under the following conditions:

- When you first turn on system power
- When you power cycle or reset the system
- When system power comes on after a power failure
- After a bugcheck (*OpenVMS*) or panic (Linux or *Tru64 UNIX*)

6.4.1 Setting the Operating System to Auto Start

The SRM auto_action environment variable determines the default action the system takes when the system is power cycled, reset, or experiences a failure.

The factory setting for **auto_action** is **halt**. The **halt** setting causes the system to stop in the SRM console. You must then boot the operating system manually.

For maximum system availability, **auto_action** can be set to **boot** or **restart**.

- With the **boot** setting, the operating system boots automatically after the SRM **init** command is issued or the Reset button is pressed.
- With the **restart** setting, the operating system boots automatically after the SRM **init** command is issued or the Reset button is pressed, and it also reboots after an operating system crash.

To set the default action to **boot**, enter the following SRM commands:

```
>>> set auto_action boot
>>> init
```

See the *AlphaServer DS15/AlphaStation DS15 Owner's Guide* for more information.

6.5 Changing the Default Boot Device

You can change the default boot device with the set bootdef_dev command.

You can designate a default boot device. You change the default boot device by using the **set bootdef_dev** SRM console command. For example, to set the boot device to the IDE CD-ROM, enter commands similar to the following:

```
>>> show bootdef_dev
bootdef_dev   dka400.4.0.1.1
>>> set bootdef_dev dqa500.5.0.1.1
>>> show bootdef_dev
bootdef_dev dqa500.5.0.1.1
```

See the *DS15 AlphaServer and DS15 AlphaStation Owner's Guide* for more information.

6.6 Setting SRM Security

The set password and set secure commands set SRM security. The login command turns off security for the current session. The clear password command returns the system to user mode.

The SRM console has two modes, user mode and secure mode.

- User mode allows you to use all SRM console commands. User mode is the default mode.
- Secure mode allows you to use only the **boot** and **continue** commands. The **boot** command cannot take command-line parameters when the console is in secure mode. The console boots the operating system using the environment variables stored in NVRAM (**boot_file**, **bootdef_dev**, **boot_osflags**).

Example 6–1 Set Password

```
>>> set password                                                ❶
Please enter the password:
Please enter the password again:
>>>

>>> set password                                                ❷
Please enter the password:
Please enter the password again:
Now enter the old password:
>>>

>>> set password
Please enter the password:
Password length must be between 15 and 30 characters ❸
>>>
```

❶ Setting a password. If a password has not been set and the **set password** command is issued, the console prompts for a password and verification. The password and verification are not echoed.

❷ Changing a password. If a password has been set and the **set password** command is issued, the console prompts for the new password and verification, then prompts for the old password. The password is not changed if the validation password entered does not match the existing password stored in NVRAM.

❸ The password length must be between 15 and 30 alphanumeric characters. Any characters entered after the 30th character are not stored.

Example 6–2 set secure

```
>>> set secure                          ❶
Console is secure. Please login.
```

```
>>> b dkb0                                              ❷
Console is secure - parameters are not allowed.
>>> login                                               ❸
Please enter the password:
>>> b dkb0                                              ❹
(boot dkb0.0.0.3.1)
.
.
.
```

❶ The **set secure** command console puts the console into secure mode.

❷ The operator attempts to boot the operating system with command-line parameters. A message is displayed indicating that boot parameters are not allowed when the system is in secure mode.

❸ Entering the **login** command turns off security features for the current console session.

❹ After successfully logging in, the operator enters a **boot** command with command-line parameters.

The **set secure** command enables secure mode. If no password has been set, you are prompted to set the password. Once you set a password and enter the **set secure** command, secure mode is in effect immediately and only the **continue, boot** (using the stored parameters), and **login** commands can be performed.

The syntax is:

set secure

Example 6–3 clear password

```
>>> clear password
Please enter the password:                                  ❶
Console is secure
>>> clear password
Please enter the password:                                  ❷
Password successfully cleared.
>>>
```

❶ The wrong password is entered. The system remains in secure mode.

❷ The password is successfully cleared.

The **clear password** command is used to exit secure mode and return to user mode. To use **clear password**, you must know the current password. Once you clear the password, the console is no longer secure.

To clear the password without knowing the current password, you must use the **login** command in conjunction with the RMC halt in/out commands.

6.7 Configuring Devices

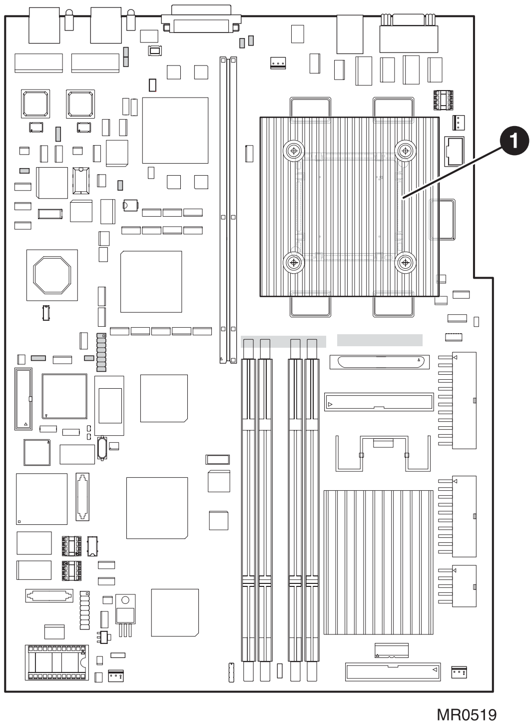
Become familiar with the configuration requirements for CPUs and memory before removing or replacing those components. See Chapter 8 for removal and replacement procedures.

WARNING: To prevent injury, access is limited to persons who have appropriate technical training and experience. Such persons are expected to understand the hazards of working within this equipment and take measures to minimize danger to themselves or others. These measures include:

- **Remove any jewelry that may conduct electricity.**
- **If accessing the system card cage, power down the system and wait 2 minutes to allow components to cool.**
- **Wear an anti-static wrist strap when handling internal components.**

6.7.1 CPU Location

Figure 6–1 CPU Location



The CPU ❶ is located on the main logic board.

6.7.2 Memory Configuration

Become familiar with the rules for memory configuration before adding DIMMs to the system.

Refer to Figure 6–3 and observe the following rules for installing DIMMs.

- You can install up to 4 DIMMs. There are two memory arrays (0 and 2). An array consists of 2 DIMMs, which must be the same capacity and type.
- A maximum of 4 GB of memory is supported.
- A memory array must be populated with two DIMMs of the same size and speed. (See the following table for supported sizes and capacity.)
- Populate memory arrays in numerical order, starting with array 0.

CAUTION: *Using different DIMMs in an array may result in loss of data.*

The following table describes all the supported DIMM configurations.

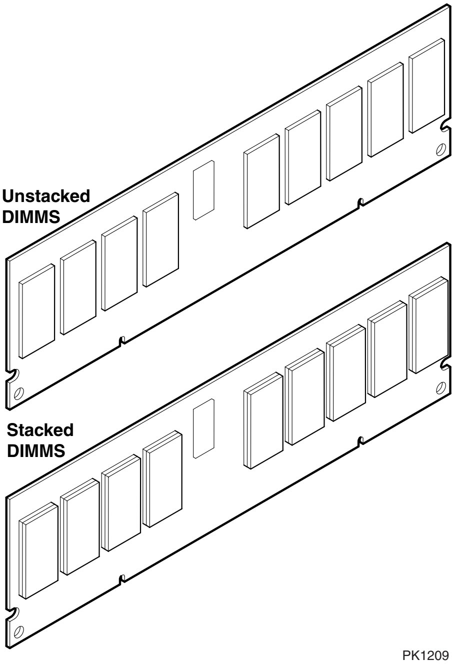
Table 4–5 Supported DIMM Configurations

Total Memory	DIMM0 J14	DIMM2 J15	DIMM1 J12	DIMM3 J13	Remarks
512MB	256MB		256MB		minimum allowed configuration
1024MB	256MB	256MB	256MB	256MB	recommended for performance
1024MB	512MB		512MB		
1536MB	256MB	512MB	256MB	512MB	
1536MB	512MB	256MB	512MB	256MB	
2048MB	512MB	512MB	512MB	512MB	recommended for performance
2048MB	1024MB		1024MB		
2560MB	1024MB	256MB	1024MB	256MB	
2560MB	256MB	1024MB	256MB	1024MB	
3072MB	1024MB	512MB	1024MB	512MB	
3072MB	512MB	1024MB	512MB	1024MB	
4096MB	1024MB	1024MB	1024MB	1024MB	recommended for performance

DIMM Information for Two System Types

You can mix stacked and unstacked DIMMs within the system, but not within an array. The DIMMs within an array must be of the same capacity and type (stacked or unstacked) because of different memory addressing.

Figure 6–2 Stacked and Unstacked DIMMs



Only the following DIMMs and DIMM options can be used in the DS15 system.

Density	DIMM	DIMM Option (2 DIMMs per)
512 MB	20-01EBA-09	3X-MS315-EA
1 GB	20-00FBA-09	3X-MS315-FA
2 GB	20-00GBA-09	3X-MS315-GA

CAUTION: *Using different DIMMs may result in loss of data.*

Memory Performance Considerations

Interleaved operations reduce the average latency and increase the memory throughput over non-interleaved operations. With one memory option (2 DIMMs) installed, memory interleaving will not occur. For 2-way interleaving, array 0 and 2 must have the same size memory.

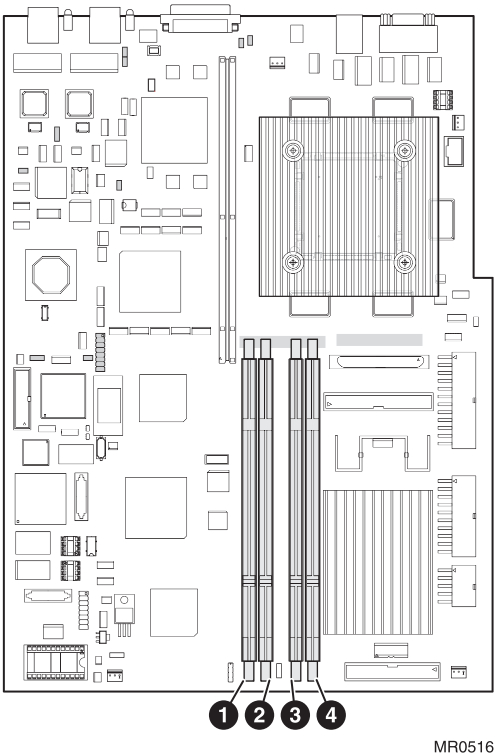
The output of the **show memory** command provides the memory interleaving status of the system.

```
 Memory
  Array        Size        Base Address     Intlv Mode
---------  ----------  ----------------  ----------
    0        1024Mb     0000000000000000     2-Way
    2        1024Mb     0000000040000000     2-Way

     2048 MB of System Memory
```

Populate both arrays with the same size memory. See Figure 6–3 for array locations.

Figure 6–3 Memory Configuration



❶ Memory DIMM slot - array 0, DIMM 1

❷ Memory DIMM slot - array 2, DIMM 3

❸ Memory DIMM slot - array 0, DIMM 0

❹ Memory DIMM slot - array 2, DIMM 2

6.7.3 PCI Configuration and Installation

The DS15 PCI slots are all 3.3 volts, and are normally automatically configured when you boot the system after installing the option.

When installing PCI option modules, you do not normally need to perform any configuration procedures; the system configures PCI modules automatically. But because some PCI option modules require and provide their own configuration utility CDs, refer to the option documentation.

PCI Slot Information

PCI slot 1 is the bottom slot on a desktop or rackmounted system or the right-hand slot as viewed from the back of a pedestal system.

PCI option modules are either designed for 5.0 volts or 3.3 volts, or are universal in design and can plug into either 3.3 or 5.0 volt slots. The DS15 system provides only 3.3 volt slots.

Some PCI options require drivers to be installed and configured. These options come with a CD-ROM. Refer to the installation document that came with the option and follow the manufacturer's instructions.

There is no direct correspondence between the physical numbers of the slots on the PCI riser and the logical slot identification reported with the SRM console **show config** command. Table 6-2 maps the physical slot numbers to the SRM logical ID numbers for the PCI slots.

Table 6-2 Comparison of Physical and Logical Slot Numbering

Physical Slot Number	Hose Number	Logical Slot ID
1	2	7
2	2	8
3	2	9
4	2	10

PCI Configuration Rules

To run at 66 MHz, the following conditions must be met:

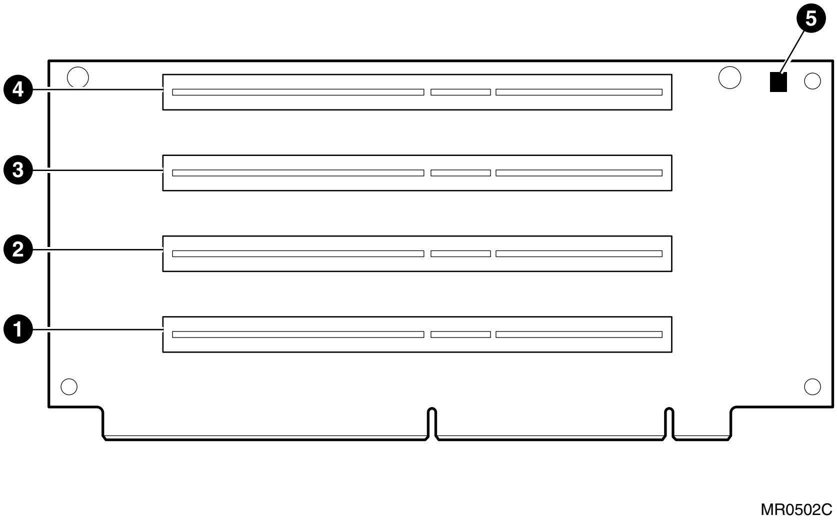
- Both slot 3 or 4 must be empty.
- A 33 MHz module must not be installed in either slot 1 or 2.
- A 66 MHz modules must be installed in either slot 1 and/or 2, otherwise the bus will run at 33 MHz.

CAUTION: *Check the keying before you install the PCI module and do not force it in. Plugging a module into a wrong slot can damage it.*

Table 6–3 How Physical I/O Slots Map to Logical Slots

Port	Hose	Physical Slot	SRM Logical Slot
A	2	1	7
		2	8
		3	9
		4	10

Figure 6–4 Slots on the PCI Riser Card



❶ Slot 1 – 66/33MHz, 3.3v

❷ Slot 2 – 66/33MHz, 3.3v

❸ Slot 3 – 33MHz, 3.3v

❹ Slot 4 – 33MHz, 3.3v

❺ LED – connected to +5 VAUX

For more information, see http://h18002.www1.hp.com/alphaserver/ .

6.8 Booting Linux

Obtain the Linux installation document and install Linux on the system. Then verify the firmware version, boot device, and boot parameters, and issue the boot command.

The procedure for installing Linux on an Alpha system is described in the Alpha Linux installation document for your Linux distribution. The installation document can be downloaded from the following Web site:

http://www.compaq.com/alphaserver/linux

You need V6.6-24 or higher of the SRM console to install Linux. If you have a lower version of the firmware, you will need to upgrade. For instructions and the latest firmware images, see the following URL.

http://ftp.digital.com/pub/DEC/Alpha/firmware/

Linux Boot Procedure

1. Power up the system to the SRM console and enter the **show version** command to verify the firmware version.

```
>>> show version
version                    V6.6-24 Sept 5 2003 08:36:11
>>>
```

2. Enter the **show device** command to determine the unit number of the drive for your boot device, in this case dka0.0.0.8.0.

```
>>> show device
dka0.0.0.8.0               DKA0              COMPAQ BF03665A32  3B01
dka100.1.0.8.0             DKA100            COMPAQ BF03665A32  3B01
dqa0.0.0.13.0              DQA0                        DW-224E  A.1J
dva0.0.0.1000.0            DVA0*
eia0.0.0.9.0               EIA0              00-02-A5-20-C0-39
eib0.0.0.10.0              EIB0              00-02-A5-20-C0-3A
pka0.7.0.8.0               PKA0                  SCSI Bus ID 7
pkb0.7.0.108.0             PKB0                  SCSI Bus ID 7
>>>
```

* DS15 systems have no floppy drives.

3. From SRM enter the **boot** command. The following example shows **boot** output.

Example 6–4 Linux Boot Output

```
(boot dqa0.0.0.13.0)
block 0 of dqa0.0.0.13.0 is a valid boot block
reading 174 blocks from dqa0.0.0.13.0
bootstrap code read in
base = 2b6000, image_start = 0, image_bytes = 15c00(89088)
initializing HWRPB at 2000
initializing page table at 7fff0000
initializing machine state
setting affinity to the primary CPU
jumping to bootstrap code
aboot: Linux/Alpha SRM bootloader version 0.9b
aboot: switching to OSF/1 PALcode version 1.92
aboot: booting from device 'IDE 0 13 0 0 0 0 0'
aboot: no disklabel found.
Welcome to aboot 0.9b
Commands:

 h, ?                 Display this message
 q                    Halt the system and return to SRM
 p 1-8                Look in partition <num> for configuration/kernel
 l                    List preconfigured kernels
 d <dir>              List directory <dir> in current filesystem
 b <file> <args>      Boot kernel in <file> (- for raw boot)
 i <file>             Use <file> as initial ramdisk
                      with arguments <args>
 0-9                  Boot preconfiguration 0-9 (list with 'l')
aboot> l
iso: Max size:329552   Log zone size:2048
iso: First datazone:28   Root inode number 57344
#
# Red Hat Linux/Alpha aboot configuration options:
#
#   0 - Boot the Red Hat Linux installer
#   1 - Boot the Red Hat Linux installer with serial console (ttyS0)
#   2 - Boot the Red Hat Linux installer with callback console (srm)
#       (required for "serial" console on AlphaServers ES47, ES80, GS1280)
#   3 - Boot the Red Hat Linux installer in text only mode
#   4 - Boot the Red Hat Linux installer in text only rescue mode
#   5 - Boot the Red Hat Linux installer but allow manual selection of
drivers
#   6 - Boot the Red Hat Linux installer and allow for other than just
#       a CD install (offers http, nfs, ftp, and local disk install methods)
#
# Additional arguments can be provided at the aboot> prompt. For example,
# '6 console=ttyS0' will boot an 'expert' install using a serial console.
#
0:/kernels/vmlinux.gz initrd=/images/cdrom.img
1:/kernels/vmlinux.gz initrd=/images/cdrom.img console=ttyS0
2:/kernels/vmlinux.gz initrd=/images/cdrom.img console=srm



3:/kernels/vmlinux.gz initrd=/images/cdrom.img text
4:/kernels/vmlinux.gz initrd=/images/cdrom.img rescue
5:/kernels/vmlinux.gz initrd=/images/cdrom.img noprobe
6:/kernels/vmlinux.gz initrd=/images/cdrom.img expert
```

aboot>

NOTE: *The Linux banner may be slightly different on other Linux distributions.*

Chapter 7
Using the Remote Management Console

You can manage the system through the Remote Management Console (RMC). The RMC is implemented through an independent microprocessor that resides on the system board. The RMC also provides configuration and error log functionality.

This chapter explains the operation and use of the RMC. Sections are:

- RMC overview
- Operating modes
- Terminal setup
- SRM environment variables for COM1
- Entering the RMC
- Using the command-line interface
- Configuring remote dial-in and dial-out alert
- RMC firmware update and recovery
- Resetting the RMC to factory defaults
- RMC command reference
- Troubleshooting tips

7.1 RMC Overview

The remote management console provides a mechanism for monitoring the system (voltages, temperature, and fans) and manipulating it on a low level (reset, power on/off, halt).

The RMC performs monitoring and control functions to ensure the successful operation of the system.

- Monitors the thermal sensor on the system motherboard.
- Monitors voltages and fans
- Detects alert conditions such as excessive temperature, fan failure, and voltage failure. On detection, pages an operator, and sends an interrupt to SRM, which then passes the interrupt to the operating system or an application.
- Shuts down the system if any fatal conditions exist. For example:
- The temperature reaches the failure limit.
- Any system fan failure.
- Provides a command-line interface (CLI) for the user to control the system. From the CLI you can power the system on and off, halt or reset the system, and monitor the system environment.
- Passes error log information to shared RAM so that this information can be accessed by the system.

The RMC logic is implemented using the QLogic Zircon baseboard management controller. The RMC logic is responsible for monitoring temperature, fan speed, and all voltages. The RMC firmware images (booter and runtime) are stored in flash ROM. If the firmware should ever become corrupted or obsolete, you can update it manually using the Loadable Firmware Update Utility. See Chapters 2 and 5 for details. The microprocessor can also communicate with the system power control logic to turn on or turn off power to the rest of the system.

You can gain access to the RMC as long as AC power is available to the system (through the wall outlet). Thus, if the system fails, you can still access the RMC and gather information about the failure.

Configuration, Error Log, and Asset Information

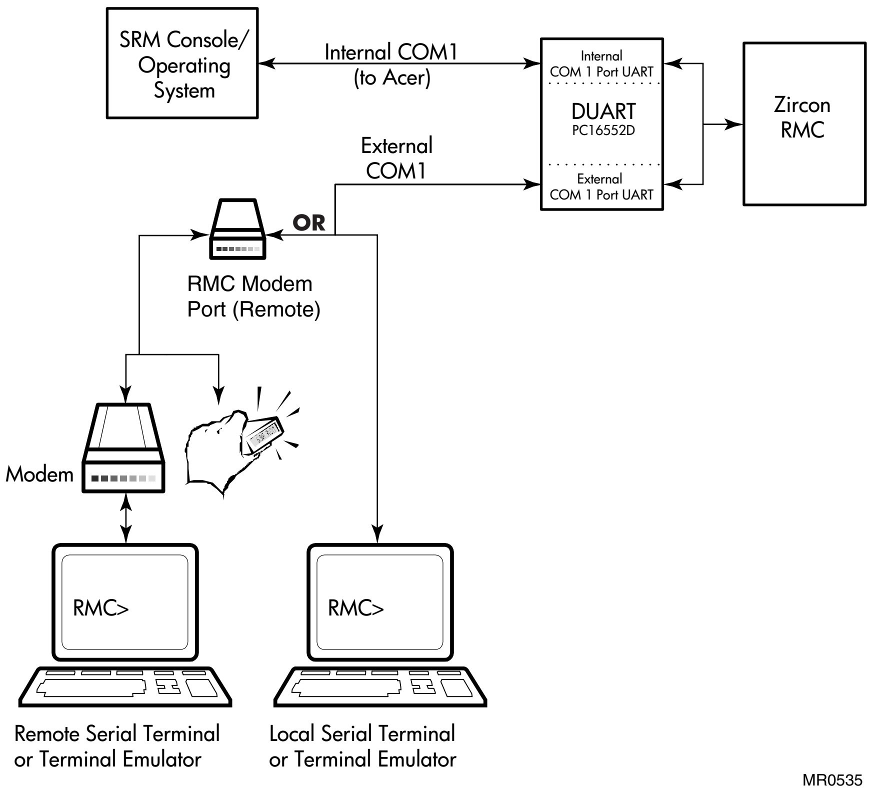
The RMC provides additional functionality to read and write configuration and error log information to Field Replaceable Unit (FRU) error log devices. These operations are carried out via shared RAM (also called dual-port RAM or DPR).

At power-on, the RMC reads the EEPROMs in the system and dumps the contents into the DPR. These EEPROMs contain configuration information, asset inventory and revision information, and error logs. During power-up the SROM sends status and error information for the CPU to the DPR. The system also writes error log information to the DPR when an error occurs. Service providers can access the contents of the DPR to diagnose system problems.

7.2 Operating Modes

The RMC can be configured to manage different data flow paths defined by the com1_mode environment variable. In Through mode (the default), all data and control signals flow from the system COM1 port through the RMC to the active external port. You can also set bypass modes so that the signals partially or completely bypass the RMC. The com1_mode environment variable can be set from either SRM or the RMC. See Section 7.11.

Figure 7–1 Data Flow in Through Mode



Through Mode

Through mode is the default operating mode. The RMC routes every character of data between the internal system COM1 port and the external COM1 port. If a modem is connected, the data goes to the modem. The RMC filters the data for a specific escape sequence. If it detects the escape sequence, it enters the RMC CLI.

Figure 7–1 illustrates the data flow in Through mode. The internal system COM1 port is connected to one port of the DUART chip, and the other port is connected to a 9-pin external COM1, providing full modem controls. The DUART is controlled by the RMC microprocessor, which moves characters between the two UART ports. The escape sequence signals the RMC to enter the CLI. Data issued from the CLI is transmitted between the RMC microprocessor and the external port.

In Through mode, the RMC also broadcasts power-up and power-down error messages through the COM1 port. Additional RMC broadcast messages may occur when the RMC CLI is active.

NOTE: *The internal system COM1 port should not be confused with the external COM1 serial port on the back of the system.*

7.2.1 Bypass Modes

For modem connection, you can set the operating mode so that data and control signals partially or completely bypass the RMC. The bypass modes are Snoop, Soft Bypass, and Firm Bypass.

Figure 7–2 Data Flow in Bypass Mode

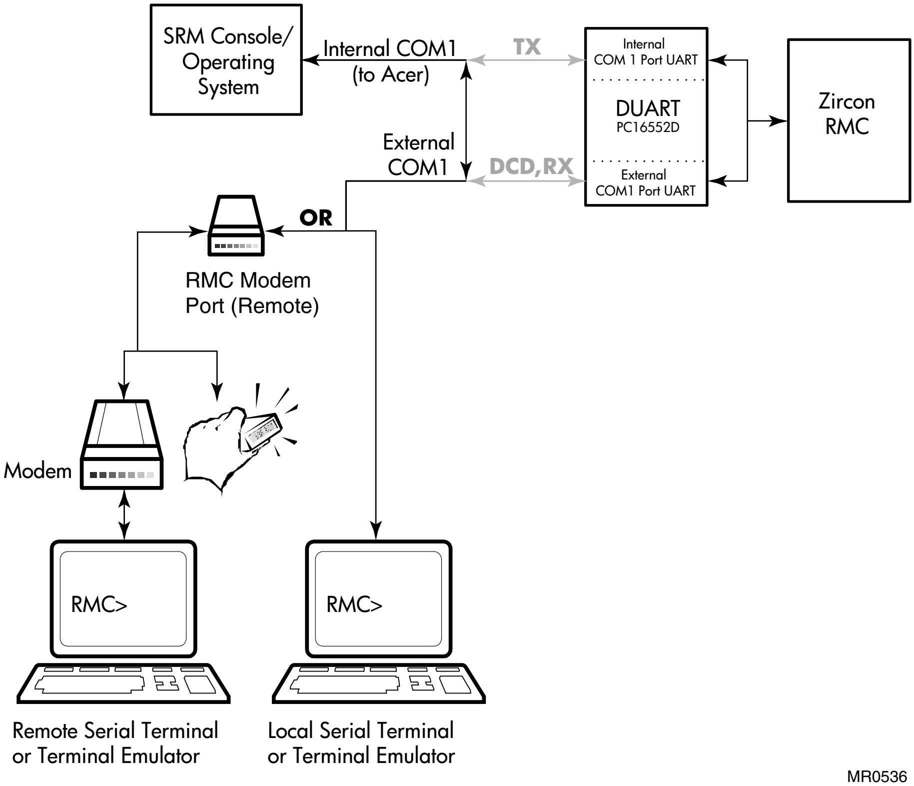


Figure 7–2 shows the data flow in the bypass modes. Note that the internal system COM1 port is connected directly to the external COM1 port.

NOTE: *You can connect a serial terminal to the external COM1 port in any of the bypass modes.*

Snoop Mode

In Snoop mode data partially bypasses the RMC. The data and control signals are routed directly between the system COM1 port and the external COM1 port, but the RMC taps into the data lines and listens passively for the RMC escape sequence. If it detects the escape sequence, it enters the RMC CLI.

The escape sequence is also passed to the system on the bypassed data lines. If you decide to change the default escape sequence, be sure to choose a unique sequence so that 1) the system software does not interpret characters intended for the RMC and 2) you ensure that you don't inadvertently invoke the RMC CLI.

In Snoop mode the RMC is responsible for configuring the modem for dial-in as well as dial-out alerts and for monitoring the modem connectivity.

Because data passes directly between system COM1 port and the 9-pin external COM1 port (bypassing the DUART), Snoop mode is useful when you want to monitor the system but also ensure optimum COM1 performance.

In Snoop mode, the RMC also broadcasts power-up and power-down error messages through the COM1 port. Additional RMC broadcast messages may occur when the RMC CLI is active.

Soft Bypass Mode

In Soft Bypass mode all data and control signals are routed directly between the system COM1 port and the external COM1 port, and the RMC does not listen to the traffic on the COM1 data lines. The RMC is responsible for configuring the modem and monitoring the modem connectivity. If the RMC detects loss of carrier or the system loses power, it switches automatically into Snoop mode. If you have set up the dial-out alert feature, the RMC pages the operator if an alert is detected and the modem line is not in use.

Soft Bypass mode is useful if management applications need the COM1 channel to perform a binary download, because it ensures that RMC does not accidentally interpret some binary data as the escape sequence.

After downloading binary files, you can set the **com1_mode** environment variable from the SRM console to switch back to Snoop mode or other modes for accessing the RMC. The RMC will also switch back to Snoop mode when the system power is off or when no DCD signal is detected on COM1.

Firm Bypass Mode

In Firm Bypass mode all data and control signals are routed directly between the system COM1 port and the external COM1 port. The RMC does not configure or monitor the modem. Firm Bypass mode is useful if you want the system, not the RMC, to fully control the modem and you want to disable RMC remote management features such as remote dial-in and dial-out alert.

You can switch to other modes by resetting the **com1_mode** environment variable from the SRM console, but you must set up the RMC again from the local terminal.

7.3 Terminal Setup

Figure 7–3 and Figure 7–4 show the connections for a VT terminal and a VGA monitor to the system. To set up the RMC to monitor a system remotely, see Section 7.7 for the procedure.

Figure 7–3 Setup for RMC with VT Terminal

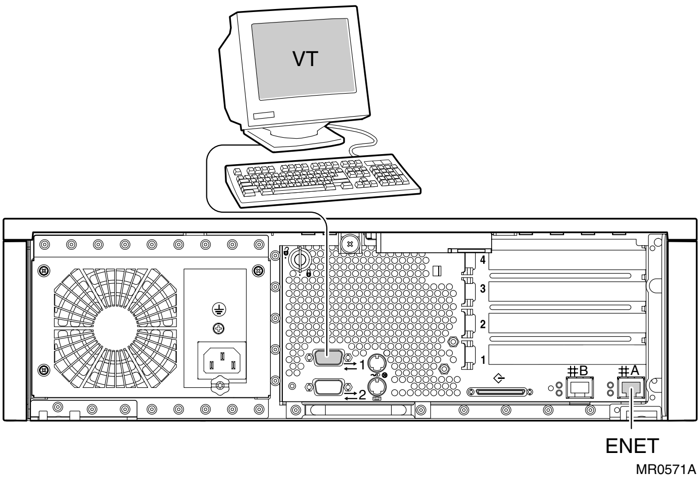
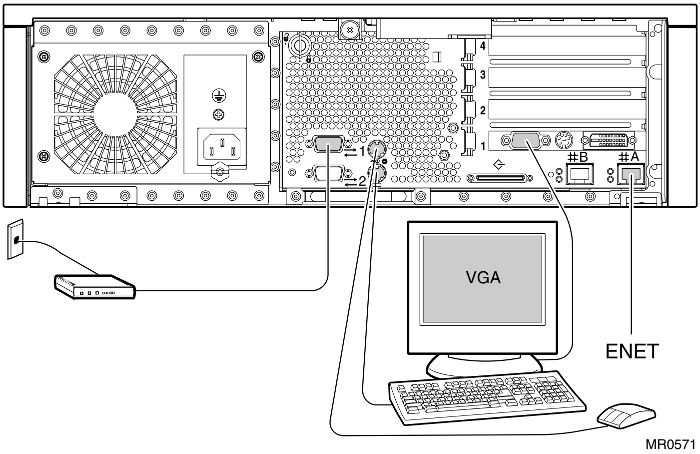


Figure 7–4 Setup for RMC with VGA Monitor



7.4 SRM Environment Variables for COM1

Several SRM environment variables allow you to set up the COM1 serial port for use with the RMC.

You may need to set the following environment variables from the SRM console, depending on how you decide to set up the RMC.

com1_baud Sets the baud rate of the COM1 serial port. The default is **9600**. See Table 6-1.

com1_flow Specifies the flow control on the serial port. The default is **software**.

com1_mode Specifies the COM1 data flow paths so that data either flows through the RMC or bypasses it. This environment variable can be set from either the SRM or the RMC. The default for com1_mode is **through**. See Section 7.11.

com1_modem Specifies to the operating system whether or not modem controls are to be utilized on COM1. The default for com1_modem is **off/disabled**.

7.5 Entering the RMC

You type an escape sequence to invoke the RMC. You can enter RMC from any of the following: Modem or terminal connected to the 9-pin external COM1 port or the local VGA monitor through the SRM console.

- You can enter the RMC from the 9-pin external COM1 port if the RMC is in Through mode or Snoop mode. In Snoop mode the escape sequence is passed to the system and displayed.
- You can enter the RMC from the local VGA monitor if COM1_MODE is set to THROUGH mode, the console environment variable is set to graphics, the 9-pin external COM1 port is inactive, and the SRM is loaded.

NOTE: *Only one RMC session can be active at a time.*

Entering from a Serial Terminal

Invoke the RMC from a serial terminal by typing the following default escape sequence:

```
^[^[ rmc
```

This sequence is equivalent to typing Ctrl/left bracket, Ctrl/left bracket, rmc. On some keyboards, the Esc key functions like the Ctrl/left bracket combination.

To exit, enter the **quit** command. This action returns you to whatever you were doing on COM1 before you invoked the RMC.

```
RMC> quit
Returning to COM port
```

Entering from the Local VGA Monitor

To enter the RMC from the local VGA monitor, the **console** environment variable must be set to **graphics** and COM1_MODE must be set to THROUGH.

Invoke the SRM console on the VGA monitor and enter the **rmc** command.

```
>>>set Com1_mode through

>>> rmc
You are about to connect to the Remote Management Console.  Use the
RMC reset command or press the front panel reset button to
disconnect and to reload the SRM console.
Do you really want to continue? [y/(n)] y
Please enter the escape sequence to connect to the Remote Management
Console.
```

After you enter the escape sequence, the system enters the CLI and the RMC> prompt is displayed.

When the RMC session is completed, reset the system with the halt/reset button (if configured for reset) on the operator control panel or issue the RMC **reset** command. (Jumper J22 pins 13-14 must be inserted for the halt/reset button to operate as a reset button.)

```
RMC> reset
Returning to COM port
```

7.6 Using the Command-Line Interface

The remote management console supports setup commands and commands for managing the system. For detailed descriptions of the RMC commands, see Section 7.11.

Command Conventions

Observe the following conventions for entering RMC commands:

- Enter enough characters to distinguish the command.

> **NOTE:** *The* **reset**, **quit**, and **rmcreset** *commands are exceptions. You must enter the entire string for these commands to work.*

- For commands consisting of two words, enter the entire first word and enough characters of the second word to distinguish it from others. For example, you can enter **disable a** for **disable alert**.
- For commands that have parameters, you are prompted for the parameter.
- Use the Backspace key to erase input.
- If you enter a nonexistent command or a command that does not follow conventions, the following message is displayed:

```
*** ERROR - unknown command ***
```

7.6.1 Displaying the System Status

The RMC status command displays the system status and the current RMC settings. Table 7–1 explains the status fields. See Section 7.11 for information on the commands used to set the user-defined status fields.

```
RMC>status

                      hp AlphaServer DS15 Platform Status
RMC Runtime Firmware Revision: V0.6-5
RMC Booter Firmware Revision: V1.0-0
System Power: ON
System Halt: Deasserted
Escape Sequence: ^[^[RMC
Remote Access: Disabled
Modem RMC Defaults: Disabled     Status: Not Initialized
RMC Password: Not Set
Alerts: Disabled     Warning Alerts: Disabled
Alert Pending: NO
Latest Alert: Fan failure
Init String:
Dial String: ATD72125
Alert String: pager #
User String: there is something wrong with my DS15 system
Com1 Baud:9600 Flow:SOFTWARE  Mode:THROUGH  Modem:DISABLED  Rmc:ENABLED
Logout Timer: 10 minutes
Voltage Status: OK
Thermal Status: OK     Thermal Shutdown: Enabled
Warning Threshold:    45.00°C    Fatal/Power-Down Threshold:    50.00°C
Fan Status: OK     Fan Shutdown: Enabled
PCI Riser: Installed
POST DPR: OK     NVRAM: OK     GPIOs: OK     LM75: OK

RMC>
```

Table 7–1 Status Command Fields

Field	Meaning
RMC Runtime Firmware Revision	RMC runtime firmware revision
RMC Booter Firmware Revision	RMC booter firmware revision
System Power	State of system power: ON = System is on. OFF = System is off.
System Halt	System halt state: Asserted = Halt is asserted Deasserted = Halt is not asserted
Escape Sequence	Current escape sequence used to access the RMC
Remote Access	Remote access state: Enabled = System is enabled for remote access via modem. Disabled = System is not enabled for remote access via modem.
Modem RMC Defaults	Older AlphaServer / AlphaStation modem-initialization sequence: Enabled = System is configured to append additional fixed commands to the user-supplied modem initialization string Disabled = System will not append additional fixed commands to the user-supplied modem initialization string
Modem RMC Status	Message indicating the current COM1 modem status. Messages include 'Initialized', 'Not Initialized', 'Not Present', and various modem initialization error messages.
RMC Password	Modem access password state Set = Password set for modem access. Not set = Password not set for modem access.
Alerts	Dial-out alert status: Enabled = Dial-out for sending alerts is enabled. Disabled = Dial-out for sending alerts is disabled.
Warning Alerts	Warning alert status: Enabled = System warnings will generate alerts. Disabled = System warnings will not generate alerts.
Alert Pending	Alert pending status: YES = Alert condition is awaiting delivery. NO = No alert condition is awaiting delivery.
Latest Alert	Text string that describes the last alert generated on the system.

Field	Meaning
Init String	Initialization string that was set for modem.
Dial String	Dial string that is sent to modem when an alert occurs
Alert String	Identification string to be sent to pager when an alert occurs. Usually set to phone number of alerting system.
User String	System notes supplied by the user.
COM1	State of the system's COM1 settings: COM1_BAUD: 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200, 38400, 57600 COM1_FLOW: NONE, SOFTWARE, HARDWARE, BOTH COM1_MODE: THROUGH, SNOOP, SOFT_BYPASS, FIRM_BYPASS COM1_MODEM: ENABLED, DISABLED COM1_RMC: ENABLED, DISABLED
Logout Timer	The amount of time before the RMC terminates an inactive modem connection (in minutes).
Voltage Status	Current state of system power: OK = All power is good FAIL = One or more of the system voltages has crossed fatal threshold
Thermal Status	System thermal status: OK = Thermal status is good WARNING = Thermal warning threshold has been crossed (fatal threshold has not been crossed) FAIL = Thermal fatal threshold has been crossed
Thermal Shutdown	Thermal failure shutdown status: Enabled = System will shutdown if the thermal fatal threshold is crossed Disabled = System will not shutdown if the thermal fatal threshold is crossed
Warning Threshold	The temperature at which a thermal warning is generated.
Fatal/Power-Down Threshold	The temperature at which a thermal failure is generated.
Fan Status	Current fan status: OK = All fans are good

Field	Meaning
	WARNING = One or more of the fans has crossed warning threshold (none have crossed fatal threshold) FAIL = One or more of the fans has crossed fatal threshold
Fan Shutdown	Fan failure shutdown status: Enabled = System will shutdown if a fan crosses its fatal threshold Disabled = System will not shutdown if a fan crosses its fatal threshold
PCI Riser	Indicates if the PCI Riser is installed: Installed = PCI Riser is installed Not Installed = PCI Riser is not installed
POST	Status results of various RMC power-on self tests: DPR (Dual-Port RAM): OK or FAIL NVRAM (RMC Non-volatile storage): OK or FAIL GPIOs (GPIOs/PCF8574 IO Expander): OK or FAIL LM75 (Thermal sensor): OK or FAIL

7.6.2 Displaying the System Environment

The RMC env command provides a snapshot of the system environment.

```
RMC>env

                    System Hardware and Environmental Status

System Voltages:                ➊
1.65V     :     1.66V       2.5V      :     2.49V       3.3V Bulk :     3.37V
5V Bulk   :     5.14V       12V Bulk  :    12.24V       -12V Bulk :   -12.19V
3.3Vsb    :     3.30V       5Vsb Bulk :     5.04V       2.85V (A) :     2.83V
2.85V (B) :     2.85V

System Temperature:             ➋
Inlet Air :    24.00°C
Warning Threshold:    45.00°C    Fatal/Power-Down Threshold:    50.00°C

Fan Speeds:                     ➌

System Fan: 1950RPM   PCI Fan  : 1560RPM  CPU Fan : 3450RPM  Disk Fan : 2730RPM

System Status Summary:          ➍
Voltage: OK    (System Power is ON)
Temperature: OK
Fan: OK

RMC>
```

NOTE: *If the system is configured with an internal storage cage, there is no disk fan. In this case the output will not display (Disk Fan: xxxRPM).*

➊ System Voltages

➋ System Temperature

➌ Fan Speeds

➍ System Status Summary of: system power, system temperature, and system fans.

7.6.3 Using Power On and Off, Reset, and Halt Functions

The RMC power {on, off}, halt, and reset commands perform the same functions as the buttons on the operator control panel.

Power On and Power Off

The RMC **power on** command powers the system on, and the **power off** command powers the system off. The Power button on the OCP, however, has precedence.

- If the system has been powered off with the Power button, the RMC cannot power the system on. If you enter the **power on** command, the message "Power-On Error: Cannot power on system when power button is off" is displayed, indicating that the command will have no effect.
- If the system has been powered on with the Power button, and the **power off** command is used to turn the system off, you can toggle the Power button to power the system back on.

When you issue the **power on** command, the terminal exits RMC and reconnects to the server's COM1 port.

```
RMC> power on

Returning to COM port

hp AlphaServer DS15 Remote Management Controller - Revision V1.1-0

RMC> power off

RMC>
```

Halt In and Halt Out

The **halt in** command halts the system, while the **halt out** command releases the halt. When you issue the **halt in** or **halt out** command, the terminal exits RMC, and reconnects to the server's COM1 port.

Toggling the Power button on the operator control panel overrides the **halt in** condition.

```
hp AlphaServer DS15 Remote Management Controller - Revision V1.1-0

RMC>halt in

Returning to COM port

hp AlphaServer DS15 Remote Management Controller - Revision V1.1-0

RMC>halt out

Returning to COM port
```

NOTE: *The SRM will not boot any images with halt asserted (**halt in**).*

Halt

The **halt** command halts the system. This is the same as pressing the halt/reset button (when configured for halt, which is the default). Jumper J22 pins 13-14 must *not* be inserted for the halt/reset button to operate as a halt button.

```
RMC>halt

Returning to COM port
```

Reset

The RMC **reset** command restarts the system. The terminal exits RMC and reconnects to the server's COM1 port.

```
RMC> reset
Returning to COM port
```

RMCReset

The **rmcreset** command resets the RMC controller. It does not reset the system.

7.7 Configuring Remote Dial-In

Before you can dial in through the RMC modem port or enable the system to call out in response to system alerts, you must configure RMC for remote dial-in.

You can use either a VT terminal or a VGA monitor to configure the RMC for remote dial-in:

1. Connect to the RMC using either a VT terminal attached to COM1 or through the VGA monitor. See Figure 7–3 and Figure 7–4.
2. Initialize the remote dial-in configuration as shown in Example 7–1.
3. Complete one of the following:
   a. If you use a VT terminal, disconnect the terminal and connect the modem to COM1.
   b. If you are using a VGA monitor, connect the modem to COM1.

NOTE: *When configuring the system for dial-in access, com1_mode must be set so that you are able to gain access to the RMC via either the VT terminal on COM1 or the VGA monitor.*

Example 7–1 Dial-In Configuration

```
RMC>>>set password                                         ❶
RMC Password: *****
Verification: *****
RMC>set init                                               ❷
Init String: at&h2e0&c1&d0s0=2
RMC>clear alert                                            ❸
RMC>disable modemdef                                       ❹
RMC>enable remote                                          ❺

Modem will be initialized when it is detected
RMC>status                                                 ❻
              hp AlphaServer DS15 Platform Status
RMC Runtime Firmware Revision: V1.1-0
RMC Booter Firmware Revision: V1.1-0
System Power: ON
System Halt: Deasserted
Escape Sequence: ^[^[RMC
Remote Access: Enabled
Modem RMC Defaults: Disabled     Status: Not Initialized
RMC Password: Set
Alerts: Disabled     Warning Alerts: Disabled
Alert Pending: NO
```

```
Latest Alert: AC Loss
Init String: AT&H2E0&C1&D0S0=2
Dial String: ATD915085554444
Alert String: ,,,,,,,,,,5551234
User String:
Com1 Baud:9600  Flow:SOFTWARE  Mode:THROUGH  Modem:DISABLED  Rmc:ENABLED
Logout Timer: 20 minutes
Voltage Status: OK
Thermal Status: OK     Thermal Shutdown: Enabled
Warning Threshold:    45.00°C    Fatal/Power-Down Threshold:    50.00°C
Fan Status: OK     Fan Shutdown: Enabled
PCI Riser: Installed
POST DPR: OK     NVRAM: OK     GPIOs: OK     LM75: OK

RMC>
```

❶ Sets the password that is prompted for at the beginning of a modem session. The string cannot exceed 14 characters and is not case sensitive. For security, the password is not echoed on the screen. When prompted for verification, type the password again.

❷ Sets the initialization string. The string is limited to 31 characters and can be modified depending on the type of modem used. Because the modem commands disallow mixed cases, the RMC automatically converts all alphabetic characters entered in the init string to uppercase.

❸ Clears the current alert.

❹ Tells the RMC not to append its own fixed flow-control and carrier-detect commands to the user-supplied modem initialization string. Instead, these will be included as part of the user-supplied initialization string.

❺ Enables remote access to the RMC modem port by configuring the modem with the setting stored in the initialization string once the modem is connected to the system.

❻ Status of the RMC configuration.

NOTE: *Once the RMC is configured, disconnect the VT terminal from COM1 (if present) and connect the modem.*

Dialing In

This example shows the screen output when a modem connection is established.

```
ATDT915085553333
CONNECT 9600/ARQ/V34/LAPM

RMC Password: *****


Welcome to RMC V1.1-0

>>>
>>>
hp AlphaServer DS15 Remote Management Controller - Revision V1.1-0

RMC>stat

                      hp AlphaServer DS15 Platform Status
RMC Runtime Firmware Revision: V1.1-0
RMC Booter Firmware Revision: V1.1-0
System Power: ON
System Halt: Deasserted
Escape Sequence: ^[^[RMC
Remote Access: Enabled
Modem RMC Defaults: Disabled     Status: Initialized
RMC Password: Set
Alerts: Disabled     Warning Alerts: Disabled
Alert Pending: NO
Latest Alert: AC Loss
Init String: AT&H2E0&C1&D0S0=2
Dial String: ATD915085554444
Alert String: ,,,,,,,,,,5551234
User String:
Com1 Baud:9600  Flow:SOFTWARE  Mode:THROUGH  Modem:DISABLED  Rmc:ENABLED
Logout Timer: 20 minutes
Voltage Status: OK
Thermal Status: OK     Thermal Shutdown: Enabled
Warning Threshold:    45.00°C    Fatal/Power-Down Threshold:    50.00°C
Fan Status: OK     Fan Shutdown: Enabled
PCI Riser: Installed
POST DPR: OK     NVRAM: OK     GPIOs: OK     LM75: OK

RMC>hangup
+++
NO CARRIER
```

At the RMC> prompt, enter commands to monitor and control the remote system.

When you have finished a modem session, enter the **hangup** command to cleanly terminate the session and disconnect from the server.

Unsetting the password

If the password is forgotten, you can reset it by using the **set password** command.

1. Enter the **set password** command at the RMC prompt.

2. Intentionally type in an incorrect verification password.
3. The following appears:
   *** ERROR – Password verification failed (Password is NOT set) ***

NOTE: *You also may reset RMC to use factory defaults. See section 7.10.*

Example 7–2 Unsetting the Password

```
RMC> set password
RMC Password: ****
Verification: ******

*** ERROR - Password verification failed (Password is NOT set) ***
```

Modem Initialization Commands

The modem initialization commands in the following table do not necessarily apply to all modems because different modems use different command sets. Consult the user's guide for your modem when determining the modem initialization string for your system configuration.

Table 7–2 Modem Initialization Commands

Modem Command	Description
&H*x*	Flow control, where *x* is as follows: 0: No flow control 1: Hardware flow control 2: Software (XON/XOFF) flow control 3: Both hardware and software flow control
E0	Local echo off
&C1	Normal Carrier Detect (CD) operations
&D0	DTR override
S0=2	Auto answer after 2 rings

7.8 Configuring Dial-Out Alert

When you are not monitoring the system from a modem connection, you can use the RMC dial-out alert feature to remain informed of system status. If dial-out alert is enabled, and the RMC detects alarm conditions within the managed system, it can call a preset pager number.

You must configure remote dial-in for the dial-out feature to be enabled. See Section 7.7.

To set up the dial-out alert feature, enter the RMC from the COM1 serial terminal or local VGA monitor.

Example 7–3 Dial-Out Alert Configuration

```
RMC>set dial                              ❶
Dial String: atd915085554444

RMC>set alert                             ❷
Alert String: ,,,,,,,,,,5551234

RMC>enable remote                         ❸

Modem will be initialized when it is detected

RMC>clear alert                           ❹

RMC>enable alert                          ❺

RMC>send alert                            ❻

RMC>status                                ❼

                      hp AlphaServer DS15 Platform Status
RMC Runtime Firmware Revision: V1.1-0
RMC Booter Firmware Revision: V1.1-0
System Power: ON
System Halt: Deasserted
Escape Sequence: ^[^[RMC
Remote Access: Enabled
Modem RMC Defaults: Disabled     Status: Not Initialized
RMC Password: Set
Alerts: Enabled     Warning Alerts: Disabled
Alert Pending: YES
Latest Alert: Test alert generated by user request
Init String: AT&H2E0&C1&D0S0=2
Dial String: ATD915085554444
Alert String: ,,,,,,,,,,5551234
```

```
User String:
Com1 Baud:9600  Flow:SOFTWARE  Mode:THROUGH  Modem:DISABLED
Rmc:ENABLED
Logout Timer: 20 minutes
Voltage Status: OK
Thermal Status: OK     Thermal Shutdown: Enabled
Warning Threshold:    45.00°C    Fatal/Power-Down Threshold:
50.00°C
Fan Status: OK     Fan Shutdown: Enabled
PCI Riser: Installed
POST DPR: OK     NVRAM: OK     GPIOs: OK     LM75: OK

RMC>
```

A typical alert situation might be as follows:

- The RMC detects an alarm condition, such as over temperature failure.
- The RMC dials your pager and sends a message identifying the system.
- You dial the system from a remote serial terminal.
- You enter the RMC, check system status with the **env** command, and, if the situation requires, power down the managed system. (In many cases, a failure may have already powered the system down.)
- When the problem is resolved, you power up and reboot the system.

The elements of the sample dial string and alert string are shown in Table 7–3. Paging services vary, so you need to become familiar with the options provided by the paging service you will be using. The RMC supports only numeric messages.

➊ Sets the string to be used by the RMC to dial out when an alert condition occurs. The dial string must include the appropriate modem commands to dial the number.

➋ Sets the alert string, typically the phone number of the modem connected to the remote system. The alert string is appended after the dial string, and the combined string is sent to the modem when an alert condition is detected.

➌ Enables remote access to the RMC's modem port.

➍ Clears current alert condition

➎ Enables the RMC to page a remote system operator.

➏ Forces an alert condition. This command is used to test the setup of the dial-out alert function. It should be issued from the local serial terminal or local VGA monitor. As long as no one connects to the modem and there is no other alert pending, this alert will be sent to the pager as soon as the modem is connected to the system. If the pager does not receive the alert, re-check your setup.

➐ Status of the RMC configuration.

NOTE: *If you do not want dial-out paging enabled at this time, enter the* **disable alert** *command after you have tested the dial-out alert function. Alerts continue to be logged, but no paging occurs.*

Table 7–3 Elements of Dial String and Alert String

Dial String	
ATXDT	The dial string is case sensitive. The RMC automatically converts all alphabetic characters to uppercase. AT = Attention. X = Forces the modem to dial “blindly” (not seek the 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)
9,	The number for an outside line (in this example, 9). Enter the number for an outside line if your system requires it. , = Pause for 2 seconds.
15085553333	Phone number of the paging service.
Alert String	
,,,,,,	Each comma (,) provides a 2-second delay. In this example, a delay of 12 seconds is set to allow the paging service to answer.
5085553332#	A call-back number for the paging service. The alert string must be terminated by the pound (#) character.
;	A semicolon (;) must be used to terminate the entire string.

NOTE: *1. The above sample dial string commands are commonly used sequences that don't necessarily apply to all configurations. Because different modems use different command sets, consult the user's guide for your modem when determining the dial-string for your system configuration.*
2. The above alert string sequence, including the pound and semicolon termination characters, is not necessarily applicable to all configurations. Consult with your paging service to determine the appropriate alert string for your configuration.

7.9 RMC Firmware Update and Recovery

This section contains definitions, explanations, and examples about RMC firmware update and recovery.

Flash Accessibility

Under normal circumstances, the RMC flash part is fully write-enabled. LFU has the ability to update the firmware components contained within this part. However, write access to this flash can be completely disabled by installing the DISABLE_FLASH jumper (J21) on pins 1-2. Installing this jumper disconnects the write-enable line from the RMC to the flash part. This disables LFU (or any other utility) from modifying the contents of the flash part.

RMC Flash Update

The RMC code consists of two images - the booter image and the runtime image Firmware updates for the RMC are performed using the standard SRM Console Loadable Firmware Update (LFU) utility. The runtime image is the FW image most likely to be updated.

Updating the Booter

It is unlikely that this image will ever need to be updated. However, should it become necessary to update the booter image, that image will be included in the ‘manual’ portion of the LFU update utility. (See Example 7–4) If a booter image update is available, the revision of the image is displayed in favor of “No Update Available”.

In order to update the booter, the write enable jumper (BOOTER_ENABLE – J22 7-8) must be installed first. If this jumper is not installed, the booter image update is not allowed.

Example 7–4 Loadable Firmware Update Utility

```
Do you want to do a manual update? [y/(n)] y


                ***** Loadable Firmware Update Utility *****
-----------------------------------------------------------------------------
 Function    Description
-----------------------------------------------------------------------------
 Display     Displays the system's configuration table.
 Exit        Done exit LFU (reset).
 List        Lists the device, revision, firmware name, and update revision.
 Update      Replaces current firmware with loadable data image.
 Verify      Compares loadable and hardware images.
 ? or Help   Scrolls this function table.
-----------------------------------------------------------------------------

UPD> l

Device         Current Revision     Filename         Update Revision

FSB            X6.6-1783            fsb_fw           X6.6-1978

SRM            X6.6-1977            srm_fw           X6.6-1977

booter         V0.5-6               booter_fw        No Update Available

rt             V0.6-3               rt_fw            V0.6-3

srom           V1.0-1               srom_fw          V1.0-1

tig            1.9                  tig_fw           1.9

UPD>
```

Emergency Runtime Image Recovery

Should the RMC runtime image become corrupted or is otherwise deemed unusable, an emergency recovery mechanism has been placed in the booter. If the situation arises where this mechanism needs to be utilized, remove power (unplug) from the system and install the RMC emergency runtime image recovery jumper (J22 pins 11-12) (see Figure 7–5 which follows). Because this mode requires that the RMC be able to control com1_mode, move jumper J30 to pins 2-3.

After re-applying power to the system (plug in), the RMC comes up in emergency update mode, which utilizes only the booter image. Power the system on using the OCP button (the RMC prompt is not available).

Once at the SRM prompt, use the standard LFU mechanisms to update the runtime image. At the completion of the update, remove power (unplug) and then remove the RMC emergency runtime image recovery jumper. If jumper J30 was moved, return it to its initial position.

NOTE: *1. The booter image cannot be updated while in the emergency runtime image recovery mode.*
2. The amber LEDs on the OCP sequentially blink when updating the RMC images.
3. When the booter detects that the runtime image is corrupt, the system and fan fault LEDs will flash on and off in unison. The user must configure the system for emergency runtime image recovery to correct this problem.
4. When the user configures the system to enter emergency runtime image recovery mode by adding jumper J22 pins 11-12, all three amber lights flash on and off in unison until the FW update is started.
5. For a complete listing of OCP LED indications, see Section 2.2.2.

7.10 Resetting the RMC to Factory Defaults

If the non-default RMC escape sequence has been lost or forgotten, RMC must be reset to factory settings to restore the default escape sequence.

WARNING: To prevent injury, access is limited to persons who have appropriate technical training and experience. Such persons are expected to understand the hazards of working within this equipment and take measures to minimize danger to themselves or others.

The following procedure restores the default settings:

1. Shut down the operating system and unplug the power cord from the power supply.
2. Remove the system cover (see Chapter 4) and wait for all the internal LEDs to go out.
3. Insert the FORCE_DEFAULT jumper (J22 / pins 9 – 10) on the main logic board.
4. Re-install the system cover and plug system in. Note: you do not need to power the system on.
5. When the RMC becomes available on the external COM1 port, the defaults have been reset.
6. Unplug the power cord.
7. Remove the system cover and make sure all the internal system LEDs are not lit.
8. Remove the FORCE_DEFAULT jumper from the main logic board.
9. Re-install the system cover and plug in the system.
10. Press the power button on the OCP to turn the system On.

NOTE: *Resetting the RMC to the factory settings does not alter the personality of the system set by the RMC set systemtype command (Section 7.11).*

To set the RMC-related system jumpers to their default settings, configure as follows (see Figure 7–5 for locations):

- Feature_1 Jumper / J22 pins 13 – 14

  On – OCP halt/reset button performs reset

  Off – OCP halt/reset button performs halt **(default)**

- Feature_2 Jumper /J22 pins 11 – 12

  On – Forces RMC emergency image recovery mode

  Off – Normal operation **(default)**

- RMC_PASSTHRU Mode Jumper / J30

  No jumper – Always bypass the RMC

  1 – 2 Always pass through the RMC

  2 – 3 Normal operation **(default)**

  Note: The user selects modes through COM1_MODE.

- RMC Force_Default Jumper / J22 pins 9 – 10

  On – Forces RMC environment to default state

  Off – Normal operation **(default)**

- FORCE_DTR Jumper / J28

  On – Forces DTR

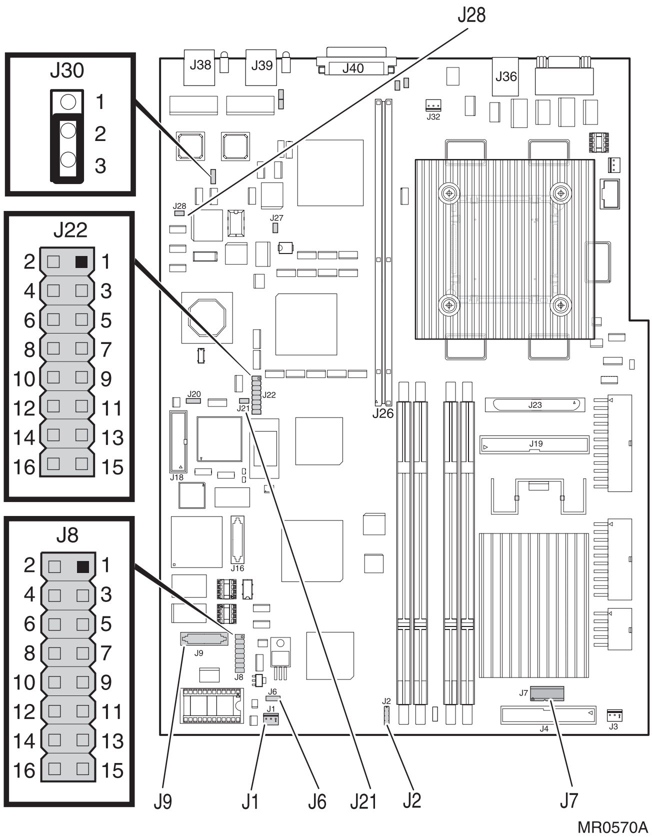
  Off – DTR unaffected **(default)**

- Booter_enable Jumper / J22 pins 7-8

  On – Allows RMC booter image updates

  Off – Disables RMC booter image updates (**default**)

Figure 7–5 RMC Jumpers (Default Positions)



7.11 RMC Command Reference

This section describes the RMC command set. Commands are listed in alphabetical order.

alert
clear {alert, log, port}
cpu
deposit
disable {alert, fan, modemdef, reboot, remote, thermal, warning, wdt}
dump
enable {alert, fan, modemdef, reboot, remote, thermal, warning, wdt}
env
fwrev
halt {in, out}
hangup
help {<optional-command-word>}
? {<optional-command-word>}
log {<optional-entry-number>}
poe
power {off, on}
quit
reset
rmcreset
send {alert}
set {alert, com1_baud, com1_flow, com1_mode, com1_modem, com1_rmc, dial, escape, init, logout, password, systemtype, user}
status

NOTE: *The* **CPU, deposit,** *and* **dump** *commands are reserved for service providers.*

alert

The **alert** command displays the latest alert condition along with detailed system status information gathered when the alert was generated.

clear alert

The **clear alert** command clears the current alert condition and causes the RMC to stop paging the system operator. If the alert is not cleared, the RMC pages the operator every 30 minutes (if the dial-out alert feature is enabled).

Once the current alert is cleared, the RMC can capture a new alert. The Alert Pending field of the **status** command becomes NO after the alert is cleared.

clear log

The **clear log** command clears all events from the system event log.

clear port

The **clear port** command clears the UARTs controlled by the RMC in an attempt to clear any 'stuck' conditions that might exist.

disable alert

The **disable alert** command disables the RMC from paging the system operator in the event that an alert condition is detected. System monitoring continues and any alert conditions that are detected will still be logged.

disable fan

The **disable fan** command disables the system from powering off in the event that a fatal fan failure occurs. By default, fan failures result in the system being powered off after a 3 minute lapse.

disable modemdef

This command instructs the RMC to use the user-supplied modem initialization string without the additional commands that were automatically appended to the initialization string on older *AlphaServer* and *AlphaStation* models.

disable reboot

The **disable reboot** command disables the watchdog timer from rebooting the system when the watchdog timer expires. By default, the system does not reboot if the watchdog timer expires.

NOTE: *The watchdog timer is not available on DS15 systems.*

---

disable remote

The **disable remote** command disables remote access to the RMC's modem port and disables automatic dial-out.

disable thermal

The **disable thermal** command disables the system from powering off in the event that a thermal failure occurs. By default, thermal failures powers off the system after a 3 minute lapse.

disable warning

When the **disable warning** command is issued, warning-level events no longer generate system alerts (this is the default state).

disable wdt

The command **disable wdt** disables the operating system watchdog timer (the default state). This does not disable the operating system from providing the watchdog clock; it simply prevents the RMC from monitoring it.

NOTE: *The watchdog timer is not available on DS15 systems.*

---

enable alert

The **enable alert** command enables the RMC to page the system operator. Before the **enable alert** command can be used, the system must be configured for remote dial-in and dial-out. See sections 7.7 and 7.8.

enable fan

The **enable fan** command allows the RMC to power off the system in the event of a fatal fan failure condition (the default state) after a 3 minute lapse.

enable modemdef

The **enable modemdef** command instructs the RMC to append additional fixed commands to the user-supplied modem initialization string. These commands were automatically appended to the initialization string on older AlphaServer / AlphaStation models. See Table 7-4 which follows.

Table 7–4 DS15 initialization commands with MODEMDEF enabled

Modem Command	Description
&C1	Normal Carrier Detect (CD) operations
&K*x*	Select flow control per the current COM1 settings, where *x* is as follows: 0: No flow control 3: Hardware flow control 4: Software (XON/XOFF) flow control 6: Both hardware and software flow control

enable reboot

The **enable reboot** command enables the watchdog timer to reset the system if the timer should expire. By default, the system does not reset if the watchdog timer expires (and the watchdog timer is enabled).

NOTE: *The watchdog timer is not available on DS15 systems.*

enable remote

The **enable remote** command enables remote access to the RMC's modem port. It also allows the RMC to automatically dial the pager number set with the **set dial** command upon the detection of an alert condition, if alerts are enabled. Before the **enable remote** command can be used, the system must be configured for remote dial-in. See section 7.7.

enable thermal

The **enable thermal** command allows the RMC to power off the system in the event of an over-temperature condition. By default, thermal failures powers off the system after a 3 minute lapse.

enable warning

The **enable warning** command allows warning-level events to generate system alerts (by default, warnings do not generate alerts).

Note that alerts are delivered in the order in which they occur. Therefore, a pending warning-level alert blocks the delivery of a fatal-level alert (although the fatal alerts continue to be logged).

enable wdt

The command **enable wdt** enables the operating system watchdog timer (disabled by default).

NOTE: *The watchdog timer is not available on DS15 systems.*

---

env

The **env** command provides a current snapshot of the status of the system environment (voltages, temperature, fans). If a sensor has crossed its warning threshold, it is displayed bold; if a sensor has crossed its fatal threshold, the reading is displayed bold and blinking.

fwrev

The **fwrev** command displays the RMC-accessible firmware revisions. Note that prior to the first successful SRM-console load, the RMC only has access to the RMC Booter image and RMC Runtime image firmware revisions.

halt

The **halt** command halts the system. This is the same as pressing and releasing the momentary contact halt button on the OCP. (Jumper J22 pins 13-14 must not be installed for the halt/reset button to operate as a halt button.)

halt in

The **halt in** command asserts halt to the system, halting the platform. To deassert a halt, issue the **halt out** command.

NOTE: *Halt will de-assert if system power is cycled.*

---

halt out

The **halt out** command releases the system from the halted state.

hangup

The **hangup** command terminates the current modem session. A modem session automatically terminates after a period of idle time set by the **set logout** command (default = 20 minutes).

help or ?

The **help** or **?** command displays the RMC command set.

help or ? command-word

Issuing the command **help** or **?** followed by the first word of another command provides additional information on all of the commands that start with the supplied word.

log

The **log** command prints out a brief summary of the last 10 system events that have been logged.

log number

Issuing the **log** command followed by a number (0-9) provides detailed information about the selected system event (0=most recent event).

poe

The **poe** command displays the latest power-on error (if any).

power off

The **power off** command performs the same function as releasing the on/off button on the OCP; it turns the system power off.

power on

The **power on** command performs the same function as pressing the on/off button on the OCP; it turns the system power on.

The system cannot be powered on with this command if the OCP power button is in the off position.

quit

The **quit** command exits the RMC and returns the terminal to external control.

reset

The **reset** command restarts the system. It performs the same function as pressing the reset button on the OCP. (Jumper J22 pins 13-14 must be inserted for the halt/reset button to operate as a reset button.)

rmcreset

The **rmcreset** command resets the RMC controller; it does *not* reset the DS15.

send alert

The **send alert** command forces an alert condition. It is used primarily to test the set-up of the dial-out alert function.

set alert

The **set alert** command sets the alert string that is transmitted through the modem when an alert condition is detected.

Generally, the alert string is set to the phone number that can be used to dial-in to the system that is experiencing the alert condition. The alert string is appended to the dial string and the combination is sent to the modem.

set com1_baud

The **set com1_baud** command is used to set the baud rate on the external 9-pin RMC/COM1 port. The available choices are: 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200, 38400, and 57600.

This command changes the setting of the SRM environment variable COM1_BAUD.

set com1_flow

The **set com1_flow** command is used to set the flow control that is to be used on the external 9-pin RMC/COM1 port. The available choices are: none, software, hardware, both.

This command changes the setting of the SRM environment variable COM1_FLOW.

set com1_mode

The **set com1_mode** command specifies the COM1 data flow path so that data either passes through the RMC or bypasses it. The available choices are: through, snoop, soft_bypass, firm_bypass.

The **set com1_mode** command changes the setting of the SRM environment variable COM1_MODE.

Com1_Mode Setting	Description
Through	All data passes through RMC and is filtered for the escape sequence that is used to enter the RMC CLI.
Snoop	Data partially bypasses RMC, but RMC taps into data lines listening for the escape sequence that is used to enter the RMC CLI.
soft_bypass	Data bypasses RMC; however, RMC automatically switches into Snoop Mode if the system is powered off or DCD is not detected.
firm_bypass	Data bypasses the RMC. You cannot gain access to the RMC CLI from this mode.

set com1_modem

The **set com1_modem** command is used to indicate whether or not modem control signals are to be used on the external 9-pin RMC/COM1 port. The available choices are: enabled or disabled. This variable is intended for use by the OS; it is not used by the RMC.

This command changes the setting of the SRM environment variable COM1_MODEM.

set com1_rmc

The **set com1_rmc** command is used to enable/disable the ability of the internal COM1 port (Acer) to access the RMC command set. After issuing the command, the user is prompted for the desired setting: enabled or disabled. This command changes the setting of the SRM environment variable COM1_RMC.

The setting of COM1_RMC is generally controlled by the SRM console; under normal circumstances, the user should not change the setting of COM1_RMC and will, therefore, not use this command.

set dial

The **set dial** command sets the string to be used by the RMC to dial out whenever an alert condition occurs. The string must be in the correct format for the attached modem. If a paging service is to be contacted, the string should include the appropriate modem commands to dial the number.

NOTE: *All lowercase characters are converted to uppercase.*

---

set escape

The **set escape** command sets a new escape sequence for invoking the RMC. The escape sequence can be any string, but cannot exceed 14 characters. A typical escape sequence includes two or more control characters.

set init

The **set init** command sets the modem initialization string. The string is limited to 31 characters and is converted to uppercase.

set logout

The **set logout** command sets the amount of time before the RMC terminates an inactive modem connection. The default is 20 minutes.

The settings are in tens of minutes (0-9). The zero (0) setting disables logout. When logout is disabled, the RMC never disconnects an idle modem session.

set password

The **set password** command lets you set or change the password at the beginning of a modem session. You must set a password to enable access through a modem. The string cannot exceed 14 characters and is not echoed to the screen.

set systemtype

The **set systemtype** command is a special hidden command that sets the current system type – *AlphaServer* DS15, *AlphaStation* DS15, or *AlphaServer* TS15. This command cannot be abbreviated – it must be typed in its entirety.

When the command is issued, it prompts for a special hard-coded password (password=setsystem15). After correctly typing the password, the user is then prompted to select the system type from a list.

NOTE: *This command is only for use by HP personnel. It does not appear in any user documentation and is not listed by the* ***help/?*** *command.*

---

set user

The **set user** command allows the user to set a user string to be displayed by the **status** command. This string is typically used to make notes about the current status of the system. The string is limited to 63 characters.

status

The **status** command displays information about the current status of the system and its RMC settings. (See section 7.6.1.)

7.12 Troubleshooting Tips

Table 7–5 lists possible causes and suggested solutions for symptoms.

Table 7–5 RMC Troubleshooting

Symptom	Possible Cause	Suggested Solution
You cannot enter the RMC from the modem.	The RMC may be in soft bypass or firm bypass mode.	Issue the **show com1_mode** command from SRM and change the setting if necessary.
The terminal cannot communicate with the RMC correctly.	System and terminal baud rates do not match.	Set the baud rate for the terminal to be the same as for the system. For first-time setup, note that the RMC and system default baud is 9600.
RMC will not answer when the modem is called.	Modem cables may be incorrectly installed.	Check modem phone lines and connections.
	RMC remote access is disabled or the modem was power cycled since last being initialized.	From the local serial terminal or VGA monitor, enter the **set password** and **set init** commands, and then enter the **enable remote** command. (See Section 7.7.)
	The modem is not configured correctly.	Modify the modem initialization string according to your modem documentation.
RMC will not answer when modem is called.	On AC power-up, RMC defers initializing the modem for 30 seconds to allow the modem to complete its internal diagnostics and initializations.	Wait 30 seconds after powering up the system and RMC before attempting to dial in.

Table 7–5 RMC Troubleshooting (Continued)

Symptom	Possible Cause	Suggested Solution
New escape sequence is forgotten.		RMC console must be reset to factory defaults.
During a remote connection, you see a “+++” string on the screen.	The modem is confirming whether the modem has really lost carrier. This is normal behavior.	
The RMC does not always display power on or power off error messages to the external RMC/COM1 port.	The display of these messages varies with the system state and the setting of com1_mode.	Set com1_mode to through mode or snoop mode.

Chapter 8
FRU Removal and Replacement

This chapter presents detailed procedures for removing and replacing Field Replaceable Units (FRUs) on *AlphaServer* DS15 systems. Unless otherwise specified, install an FRU by reversing the steps shown in the removal procedures.

8.1 Overview of FRU Procedures

The procedures are organized by relative difficulty. For example, replacing a PCI fan is easier than replacing a memory DIMM because the DIMMs are underneath the center internal storage bay. Virtually all the procedures are of the “remove and replace” style.

You can remove and replace the following components directly:

- Top cover
- Side panel
- PCI fan
- CPU fan
- Disk drive in center internal storage bay
- Front access drive (disk or tape)
- Front access storage cage
- Internal storage cage
- PCI option module

You can remove the following components only after removing one or more other components:

- PCI riser card
- Bottom disk drive of front access storage cage
- Bottom disk drive, middle drive, or half-height DVD/CD-RW drive of internal storage cage
- Power supply

- System fan
- Memory DIMM
- Operator control panel (OCP)
- Speaker
- Motherboard

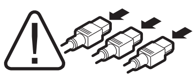
You can also refer to video procedures on the HP Intranet or order the CD.

Intranet:
http://mediadocs.mro.cpqcorp.net/video_Presentations/video%20fru/video%20fru.htm

CD:
hp AlphaServer DS15 Field Replaceable Unit (FRU) video presentation

AG-XXXXX-BE, release August 5, 2003

WARNING: To prevent injury, access is limited to persons who have appropriate technical training and experience. Such persons are expected to understand the hazards of working within this equipment and take measures to minimize danger to themselves or others. These measures include:
1. Remove any jewelry that may conduct electricity.
2. If accessing the system card cage, power down the system and wait 2 minutes to allow components to cool.
3. Wear an anti-static wrist strap when handling components.



WARNING: Before servicing the system, power it down, unplug the power cord from the power supply, and make sure that the OCP power LED on the PCI riser card is not lit. Failure to do this may result in damage of modules such as the system motherboard, and Dual Inline Memory Modules (memory DIMMs).

IMPORTANT! **After replacing FRUs and determining that the system has been restored to its normal operating condition, you must clear the system error information repository (error information logged to the DPR). Use the clear_error all command to clear all errors logged in the FRU EEPROMs and to initialize the central error repository. See Section 4.4 for details on clear_error.**

CAUTION: *Static electricity can damage integrated circuits. Always use a grounded wrist strap (29-26246) and grounded work surface when working with internal parts of a computer system.*

Remove jewelry before working on internal parts of the system.

NOTE: *If you are installing or replacing memory DIMMs or PCI modules, become familiar with the location of the module slots and configuration rules. See Chapter 6.*

8.2 Important Information before Replacing FRUs

The operating system must be shut down before you replace any FRUs. After replacing an FRU, you must clear the system error information repository with the SRM clear_error all command.

Tools

You need the following tools to remove or replace FRUs.

- Phillips #1 and #2 screwdrivers (10-inch magnetic tools are recommended)
- Flat blade screwdriver
- Cordless screwdriver
- Allen wrench (3 mm)
- Anti-static wrist strap

Hot-Plug FRUs

There are no hot-plug FRUs on the *AlphaServer* DS15.

Before Replacing FRUs

Follow the procedure below before replacing any FRUs. For universal disk drives, you must shut down the operating system, but you do not need to turn off system power.

1. Shut down the operating system.
2. Shut down power to external options, where appropriate.
3. Turn off power to the system.
4. Unplug the power cord from the power supply.

8.3 Recommended Spares

Table 8–1 lists the recommended spare parts (or FRUs) by part number and description. Figure 8–1 shows their location.

Table 8–1 Recommended Spares

Part Number	Description
12-10010-01	System Fan
12-37977-03	Lock Assembly, with Master Key
12-45971-04	Fan for Front Access Storage Cage
12-49797-01	Speaker Assembly
12-49806-04	PCI Fan
12-56450-01	CPU Fan
17-04894-03	Harness Assembly (Operator Control Panel)
17-05021-06	IDE Cable
17-05034-08	Cable Assembly (Front Access)
17-05034-11	Cable Assembly (Standard)
20-01DBA-09	256 MB DIMM 200-pin, synch, 133 MHz
20-01EBA-09	512 MB DIMM 200-pin, synch, 133 MHz
20-00FBA-09	1 GB DIMM 200-pin, synch, 133 MHz
30-10005-01	Power Supply
3R-A4412-AA	Half-Height 48x DVD/CD-RW
3X-BN46K-2E	Power Cord, US
54-30558-01	Main Logic Board (Motherboard)
54-30560-01	PCI Riser Card
70-40100-04	Internal Storage Cage
70-40481-05	Front Access Storage Cage
70-40171-04	Skins Kit
70-40204-01	Filler Bracket
70-41176-01	DS15 Enclosure
74-50910-01	Fan Guard

Table 8–2 Optional Disk and Tape Drives

The following components are not part of the recommended spares list but are included for convenient reference.

Part Number	Description
SCSI Disk Drives for Internal Storage Cage	
DS-RZ3FD-WA	36.4-GB 10,000 rpm Ultra3 SCSI drive
DS-RZ3GA-WA	72.8-GB 10,000 rpm Ultra3 SCSI drive
Universal Disk Drives for Front Access Storage Cage	
3R-A3848-AA	18.2-GB Ultra320 SCSI 15,000 rpm 1-inch Univ. disk drive
3R-A3838-AA	36.4-GB Ultra320 SCSI 10,000 rpm 1-inch Univ. disk drive
3R-A3849-AA	36.4-GB Ultra320 SCSI 15,000 rpm 1-inch Univ. disk drive
3R-A3839-AA	72.8-GB Ultra320 SCSI 10,000 rpm 1-inch Univ. disk drive
3R-A3851-AA	72.8-GB Ultra320 SCSI 15,000 rpm 1-inch Univ. disk drive
3R-A3841-AA	146-GB Ultra320 SCSI 10,000 rpm 1-inch Univ. disk drive
Tape Drives for Internal Storage Cage	
3R-A2392-AA	AIT 35/70-GB tape drive (LVD), carbon black
3R-A3752-AA	DAT 20/40-GB DDS4
3R-A3753-AA	AIT 50/100-GB
3R-A3623-AA	AIT 100/200-GB
Universal Tape Drives for Front Access Storage Cage	
3R-A2396-AA	AIT 35/70-GB LVD Univ. tape drive, uses two slots
3R-A2779-AA	AIT 50/100-GB LVD Univ. tape drive, uses two slots
3R-A2780-AA	DAT 20/40-GB DDS4 LVD Univ. tape drive, uses two slots
3R-A3621-AA	AIT 100/200-GB LVD Univ. tape drive, uses two slots

8.3.1 Power Cords

Table 8–3 lists the country-specific power cords for tower and pedestal systems.

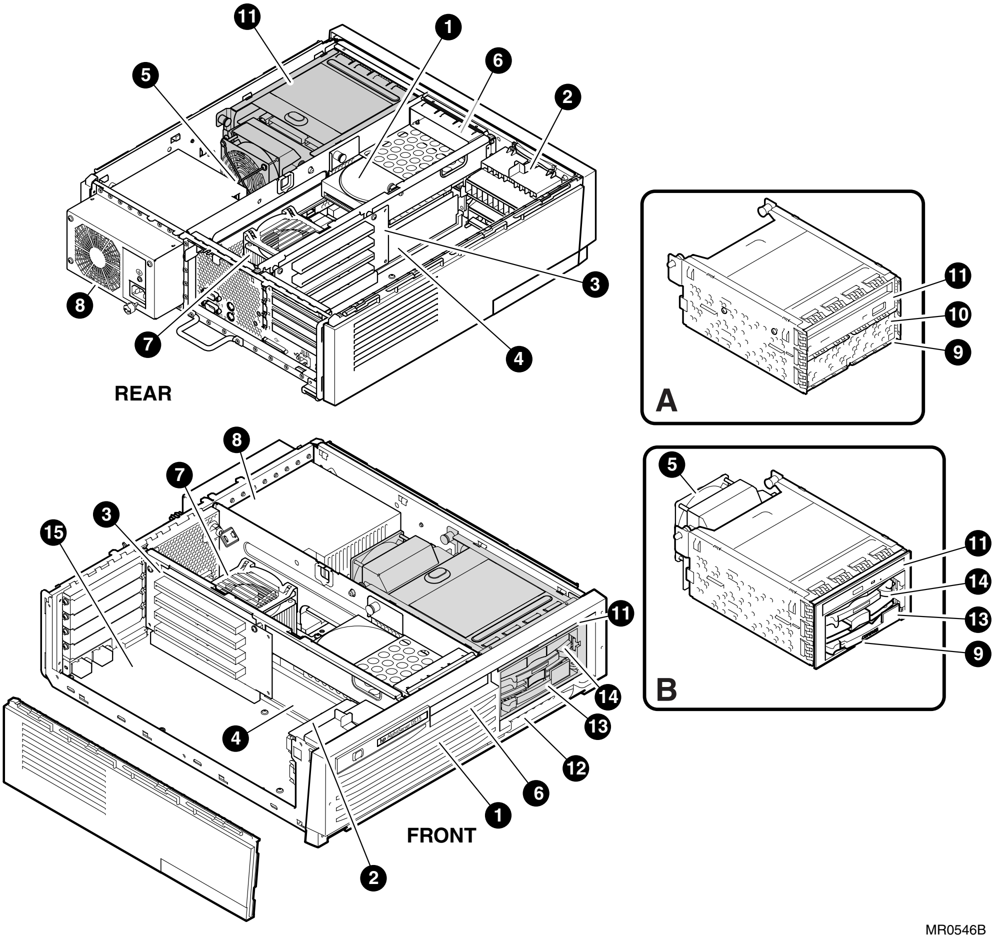
Table 8–3 Country-Specific Power Cords

Power Cord	Country	Length
Pedestal		
BN26J-1K	North American 200-240 V	75 in.
3X-BN46F-02	Japan	2.5 m
BN19H-2E	Australia, New Zealand	2.5 m
BN19C-2E	Central Europe	2.5 m
BN19A-2E	UK, Ireland	2.5 m
BN19E-2E	Switzerland	2.5 m
BN19K-2E	Denmark	2.5 m
BN19M-2E	Italy	2.5 m
BN19S-2E	Egypt, India, South Africa	2.5 m
Rackmount		
BN20Z-4E	North American 200-240 V	75 in.
BN35S	Non-US	2.5 m
3X-BN46D-4E	Japan	2.5 m

8.4 FRU Locations

Figure 8–1 shows the location of FRUs.

Figure 8–1 FRU Locations: Front and Top



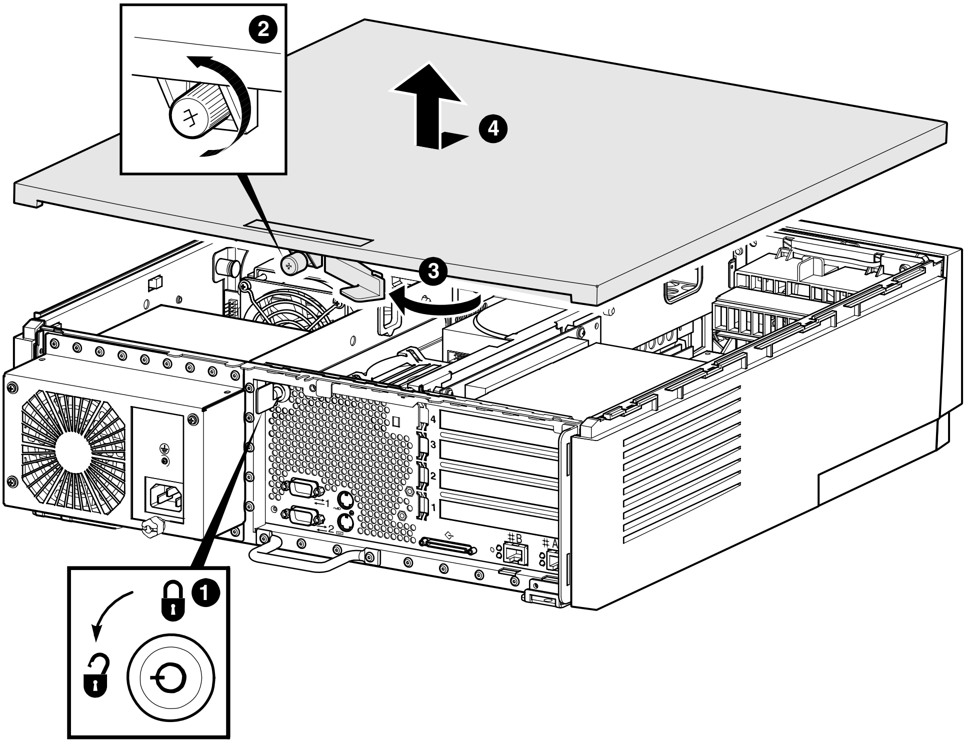
Key to Figure 8–1

❶	Center internal storage bay and disk drive
❷	PCI Fan Assembly
❸	PCI riser card
❹	Memory DIMMs
❺	Disk fan – front access storage cage only
❻	System fan
❼	CPU fan
❽	Power supply
❾	Bottom disk drive
❿	Optional disk or tape drive
⓫	DVD/CD-RW drive
⓬	Operator control panel
⓭	Front accessible disk drive
⓮	Optional front accessible disk drive
⓯	Motherboard
A	Internal storage cage
B	Front access storage cage

8.5 Removing the Top Cover

To access internal components, you must first remove the top cover. Refer to the following figure and procedure.

Figure 8–2 Removing the Top Cover



Removing the Top Cover

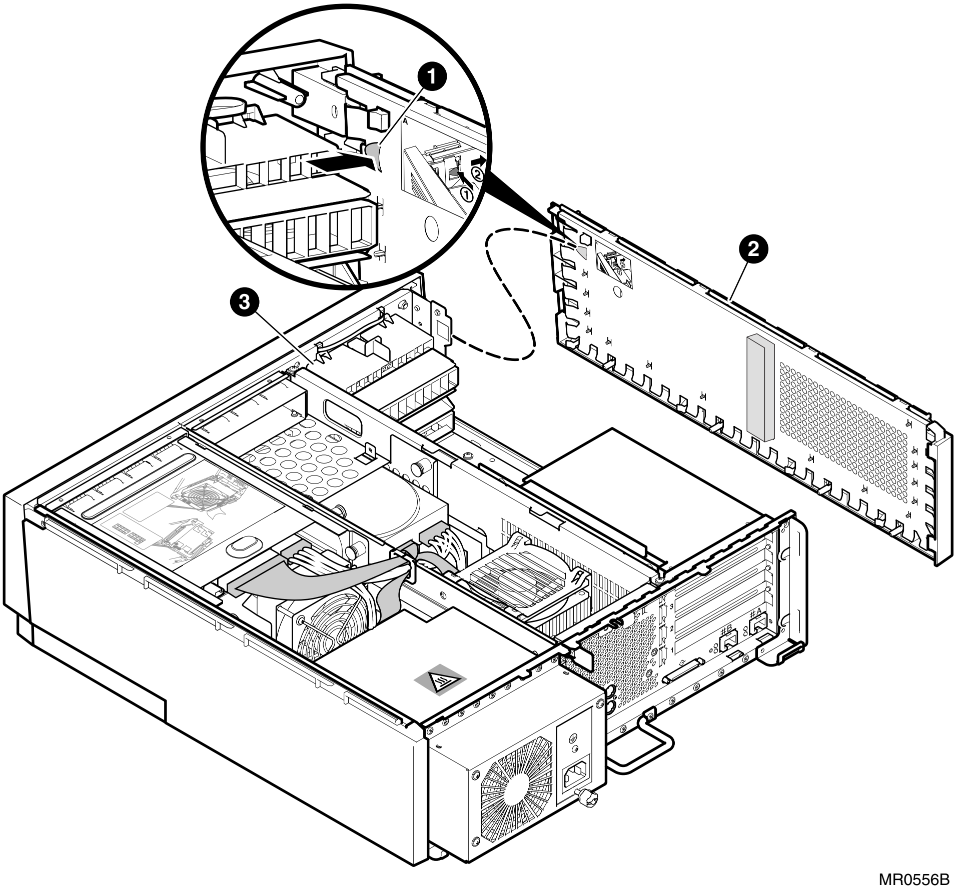
1. Unlock the system ❶ if it is locked.
2. Loosen the thumbscrew ❷ that secures the cover to the enclosure.
3. Pull the catch lever ❸ rearward to pry the cover back.
4. Slide the cover ❹ rearward and upward to remove it.
5. To replace the cover, follow these steps in reverse order.

NOTE: *Notice the quick reference labels on the inside of the top cover. The labels provide detailed information about the system.*

8.6 Removing the Side Panel

To gain access to components on the PCI side of the enclosure, you must first remove the side panel. Refer to the following figure and procedure.

Figure 8–3 Removing the Side Panel



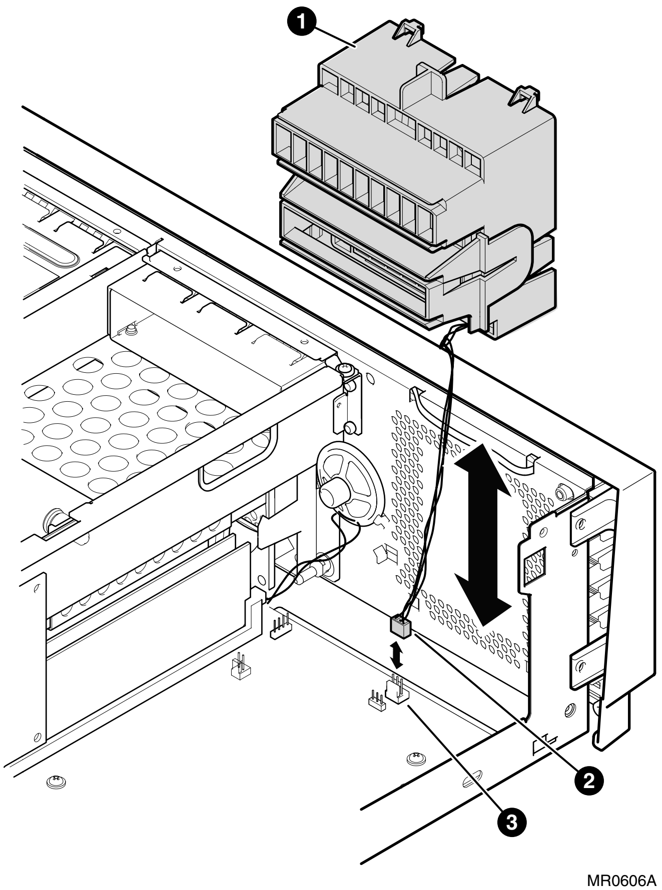
Removing the Side Panel

1. First remove the top cover as explained in Section 8.5.
2. Locate the metal tab ❶ on the panel ❷ on the PCI side of the system.
3. To allow more room for your hand, you may lift up the PCI fan ❸ and set it aside.
4. Press the metal tab, push the panel to the rear to release it, and slide it away from the system.
5. To replace the cover, follow these steps in reverse order.

8.7 Replacing the PCI Fan

The PCI fan provides cooling for the PCI side of the enclosure. Refer to the following figure and procedure when replacing the PCI fan.

Figure 8–4 Replacing the PCI Fan



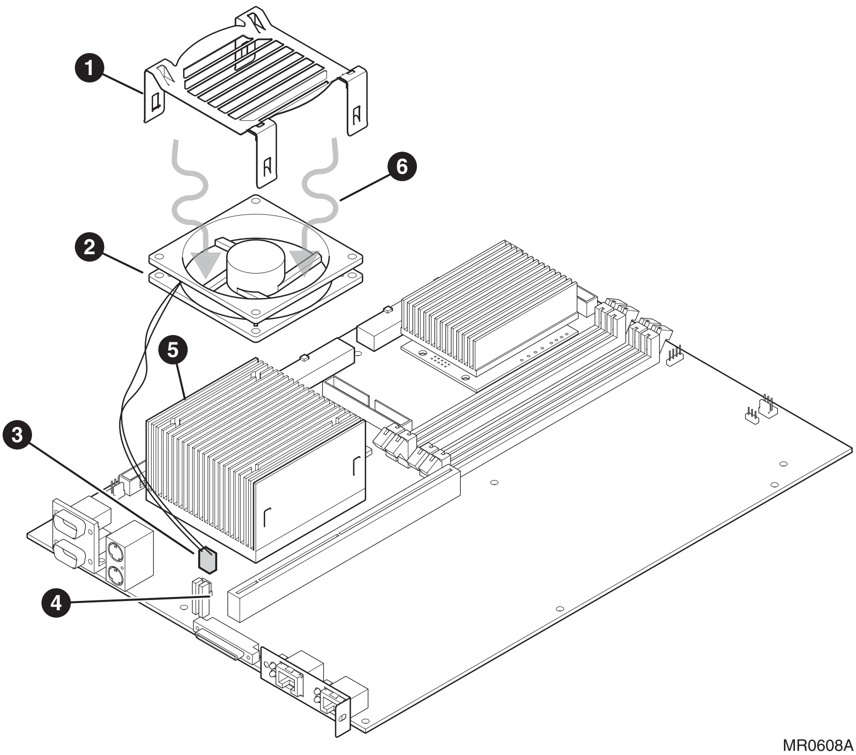
Replacing the PCI Fan

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Lift the PCI fan ❶ and lay it on its side.
5. Unplug the fan connector ❷ from the motherboard connector ❸ and lift the fan from the enclosure.
6. Plug the new fan’s connector ❷ into the motherboard connector ❸.
7. Lower the fan into place in the front of the PCI side of the enclosure. Make sure that the fan snaps into place.
8. Replace the top cover as explained in Section 8.5.

8.8 Replacing the CPU Fan

The CPU fan is mounted directly atop the CPU heat sink and provides cooling for the CPU. Refer to the following figure and procedure when replacing the CPU fan.

Figure 8–5 Replacing the CPU Fan



Replacing the CPU Fan

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the fan connector ❸ from the motherboard connector ❹.
5. Starting at a corner of the fan ❶ cover away from the center partition, press down on one of the metal retaining clips and pull it slightly away from the heat sink ❺. Repeat this action for the other clip on that same side.
6. Move to the clips next to the center partition, press down and pull each one away from the heat sink. (A flat-bladed screwdriver may be needed for these clips.)
7. Lift the fan ❷ and fan cover ❶ from the heat sink and separate the two items.
8. Place the fan cover and new fan onto the CPU heat sink so that the fan connector reaches the motherboard connector J32.

CAUTION: *Make sure the fan's airflow ❻ is down, toward the heart sink.*

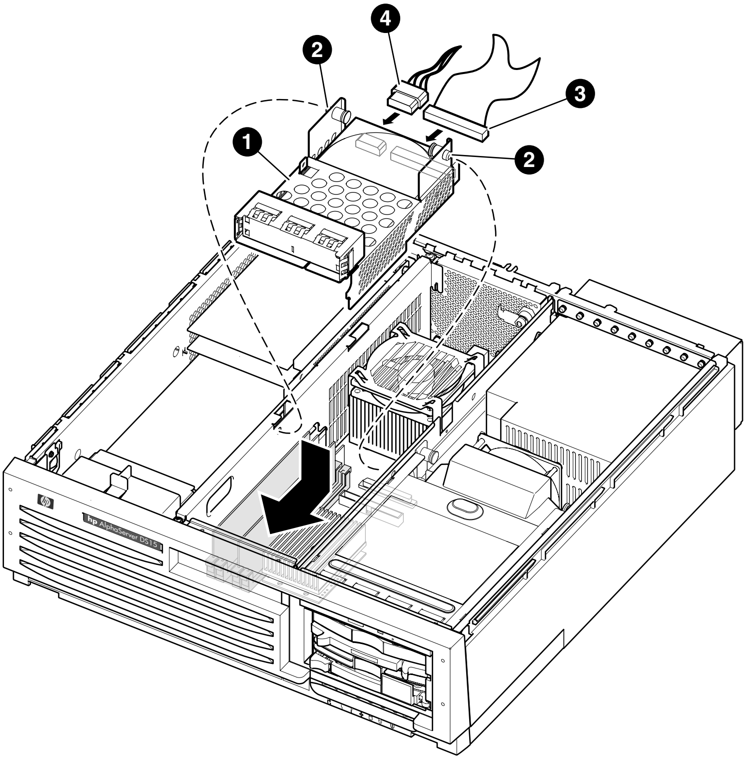
---

9. Starting at the side of the fan cover next to the center partition, press down on one of the metal retaining clips and snap the clip into place. Repeat this action for the other clip on that same side.
10. Move to the clips away from the center partition, press down on each one, and snap them into place.
11. Reconnect the fan connector ❸ to the motherboard connector ❹. The connector is keyed to install in only one way.
12. Replace the top cover as explained in Section 8.5.
13. Reconnect the power cord, turn on system power, and boot the system.

8.9 Replacing the Disk in Center Internal Storage Bay

The center internal storage bay provides a disk drive as optional storage. Refer to the following figures and procedures when replacing this disk drive.

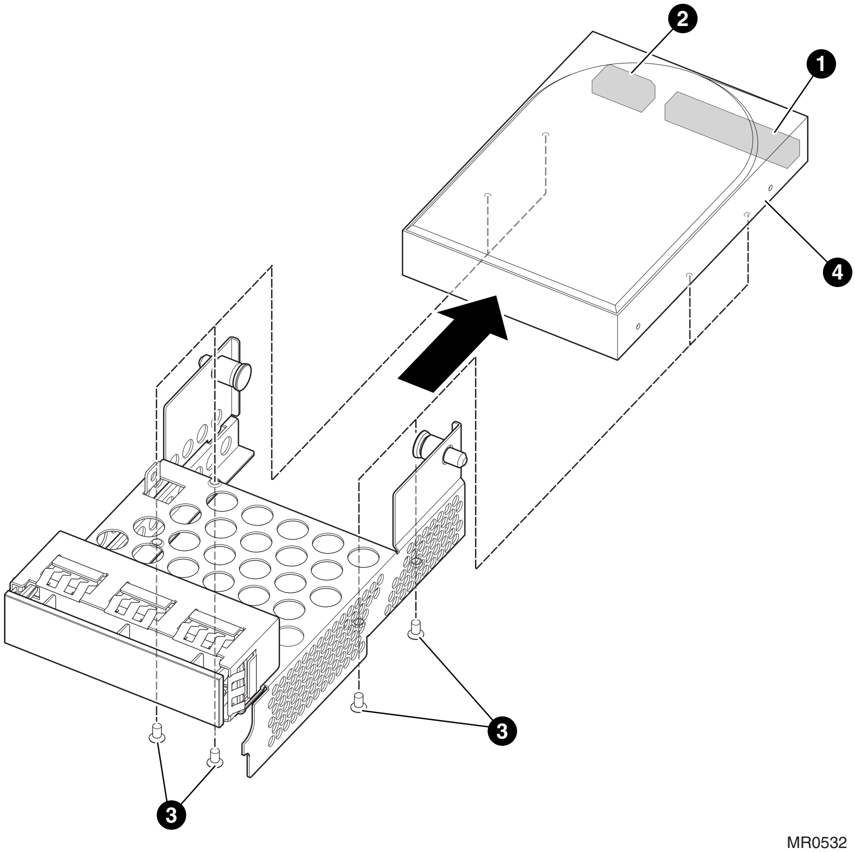
Figure 8–6 Accessing the Center Internal Storage Bay



Removing the Center Internal Storage Bay

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Pull the two spring-loaded pull pins ❷ at the rear of the storage bay ❶ and slide the unit toward the back of the enclosure.
5. Lift the storage bay from the enclosure and turn it over.
6. Remove the power cable ❹ and data cable ❸ from the rear of the storage bay and set the storage bay aside.
7. Replace the disk drive as described on the following pages.

Figure 8–7 Replacing the Disk in the Center Internal Storage Bay



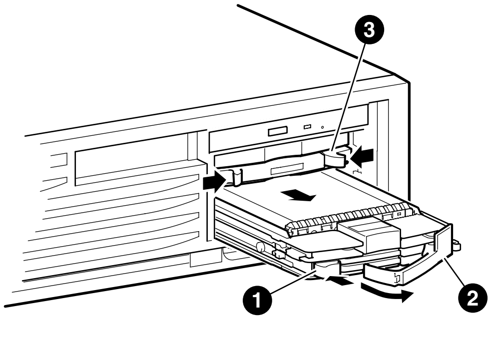
Replacing the Disk in the Center Internal Storage Bay

1. Remove the four screws ❸ from the bottom of the storage bay and slide the disk drive ❹ out.
2. Install the new disk drive and replace the bottom screws.
3. Referring to preceding Figure 8–6, connect the power cable ❸ and data cable ❹ to the new disk drive ❶.
4. Slide the storage bay forward into the enclosure.
5. Pull the two spring-loaded pull pins ❷ and lower the storage bay until the pins snap into place.
6. Replace the top cover as explained in Section 8.5.
7. Reconnect the power cord, turn on system power, and boot the system.

8.10 Replacing a Front Access Drive

If the system includes the optional front access storage cage, then one or two disk drives or one tape drive may be installed, depending on the option originally installed. Refer to the following figure and procedure when replacing a front access drive. Refer to subsequent pages when replacing a tape drive.

Figure 8–8 Replacing a Front Access Disk Drive



CAUTION: *Do not remove a drive that is in operation. Remove a drive only if its activity LED is off.*

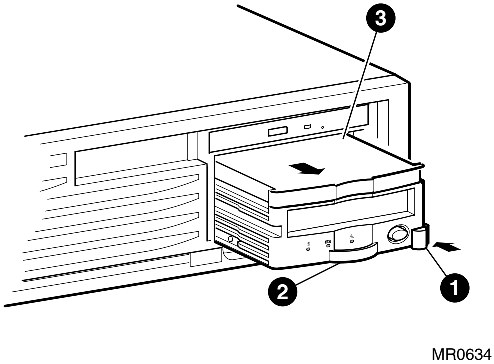
Replacing a Front Access Disk Drive

1. Verify that the disk drive is not in use (the activity LED is off).
2. To remove the drive, press in the colored rubber button ❶ to release the handle.
3. Pull the handle ❷ forward to release the SCSI connection and then pull the drive from the cage. If only one disk drive is installed, a filler plate ❸ fills the other drive slot.
4. Insert the new disk drive into the cage with the front handle ❶ fully opened. With the drive resting on top of the rail guides of the cage, slide the drive in until it stops.
5. Push in the handle ❷ to make the backplane connection and to secure the drive into the cage.

Verification

You must enter the **init** command and use the **show device** command to verify that the system sees the new drive.

Figure 8–9 Replacing a Front Access Tape Drive



CAUTION: *Do not remove a drive that is in operation. Remove a drive only if its activity LED is off.*

Replacing a Front Access Tape Drive

1. Verify that the tape drive is not in use (the activity LED is off).
2. To remove the drive, press the locking tab ❶ to the left to disengage the tape drive.
3. Pull on the handle ❷ to remove the drive from the cage. A plastic filler ❸ slides out with the tape drive.
4. Snap the new plastic filler onto the new tape drive.
5. Insert the new tape drive into the cage. With the drive resting on top of the rail guides of the cage, slide the drive in until it stops.
6. Push on the handle ❷ to make the backplane connection and verify that the locking tab ❶ snaps into place.

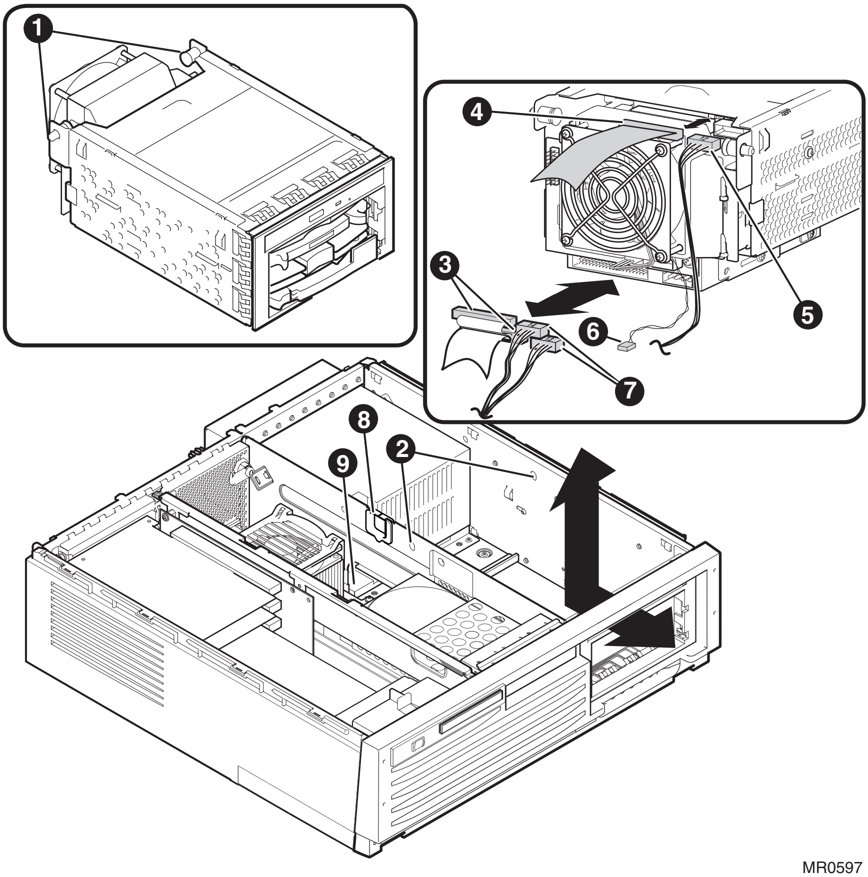
Verification

You must enter the **init** command and use the **show device** command to verify that the system sees the new drive.

8.11 Accessing the Front Access Storage Cage

The front access storage cage is an optional component and the alternative to the internal storage cage. You must remove this cage to access the bottom disk drive. Refer to the following figure and procedure when removing this cage.

Figure 8–10 Accessing the Front Access Storage Cage



Accessing the Front Access Storage Cage

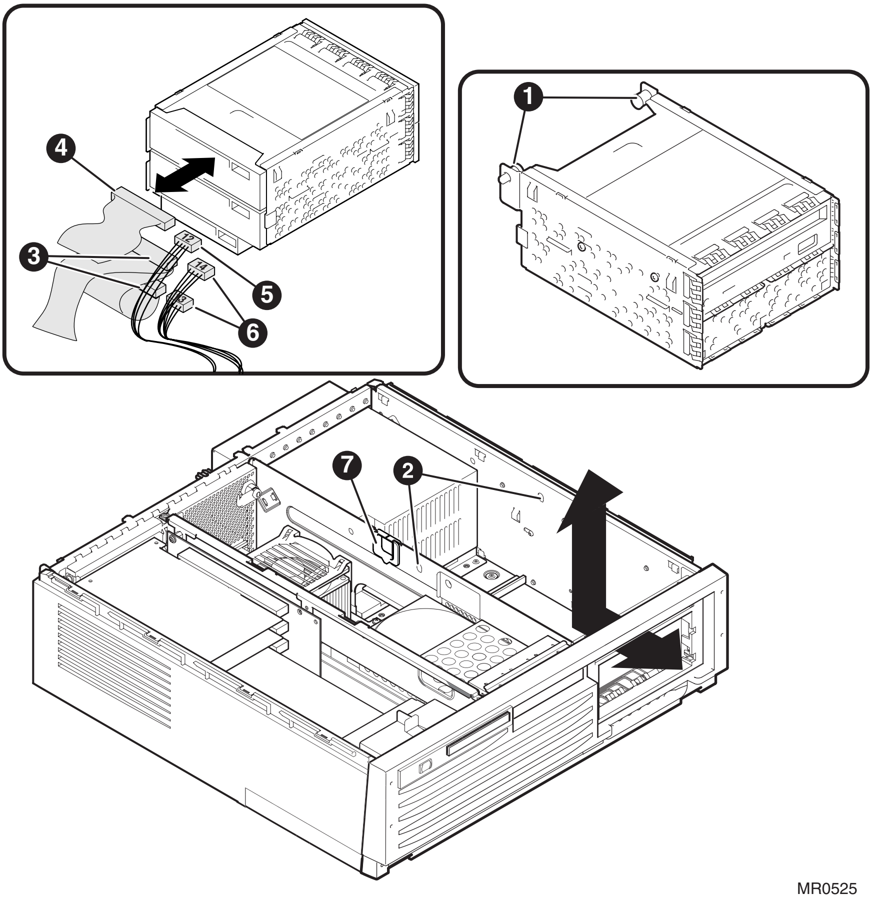
1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the IDE data cable ❹ and power cable ❺ from the back of the slimline DVD-CD-RW drive.
5. Pull the two spring-loaded posts ❶ inward so that they come out of the receiving holes ❷.
6. Pull the storage cage back, pivot the rear end up, and remove it from the enclosure. Turn the cage over to access the remaining cables.
7. Remove the SCSI cable ❸ and power cable ❼.
8. Disconnect the fan cable ❻ from the motherboard. All the cables are routed through the top slot area ❽ , except the fan cable, which is routed through the lower slot area ❾. The cage is now completely disconnected.
9. Reverse the procedure to install the storage cage.

NOTE: *The slimline DVD/CD-RW drive is not a field replaceable unit.*

8.12 Accessing the Internal Storage Cage

The internal storage cage is an optional component and the alternative to the front access storage cage. You must remove this cage to access its internal storage devices. Refer to the following figure and procedure when removing this cage.

Figure 8–11 Accessing the Internal Storage Cage



Accessing the Internal Storage Cage

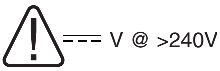
1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Pull the two spring loaded insert posts ❶ inward so that the posts come out of the receiving holes ❷.
5. Pull the storage cage back, pivot the rear end up, and remove it from the enclosure. Turn the cage over to access the remaining cables.
6. Remove the IDE data cable ❹ power cable ❺ from the DVD/CD-RW drive. All the cables are routed through the top slot area❼ .
7. Remove the SCSI cable ❸ and power cable ❻.
8. Reverse the procedure to install the storage cage.
9. Since this storage cage does not have a fan, verify that the Feature_4 jumper is installed. See Table A-2 for details.

8.13 Replacing or Installing a PCI Option Module

Some PCI option modules require drivers to be installed and configured. These options come with a CD-ROM. Refer to the installation document that came with the option module and follow the manufacturer's instructions.

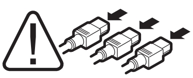
WARNING: To prevent injury, access is limited to persons who have appropriate technical training and experience. Such persons are expected to understand the hazards of working within this equipment and take measures to minimize danger to themselves or others.

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.



WARNING: High current area. Currents exceeding 240 VA can cause burns or eye injury. Avoid contact with parts or remove power prior to access.

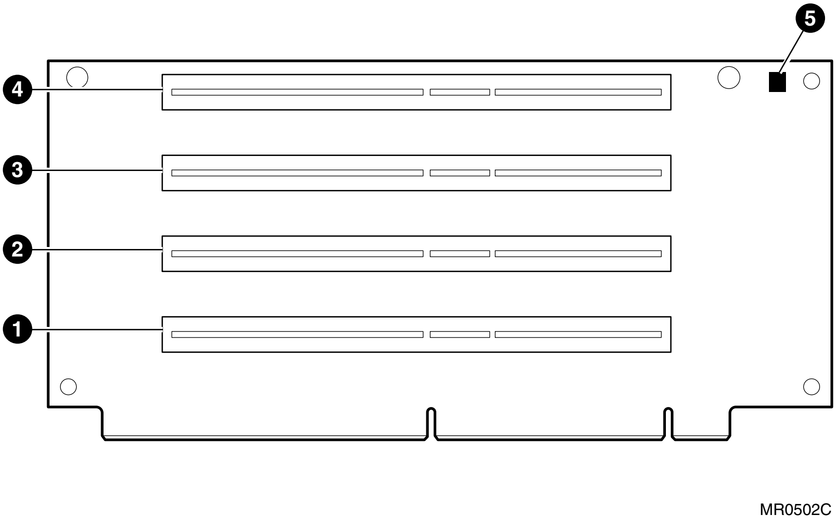
WARNING: The I/O area houses parts that operate at high temperatures. Avoid contact with components to prevent a possible burn.



WARNING: To prevent personal injury or damage to any of the system modules, unplug the power cord from the power supply before installing components. Make sure the power LEDs are not lit before removing or replacing modules.

PCI slot 1 is the bottom slot on a desktop or rackmounted system or the right-hand slot as viewed from the back of a pedestal system. The following figure shows the positions of, and other details for, the PCI slots.

Figure 8–12 Slots on the PCI Riser Card



❶ Slot 1 – 66/33 MHz, 3.3v

❷ Slot 2 – 66/33 MHz, 3.3v

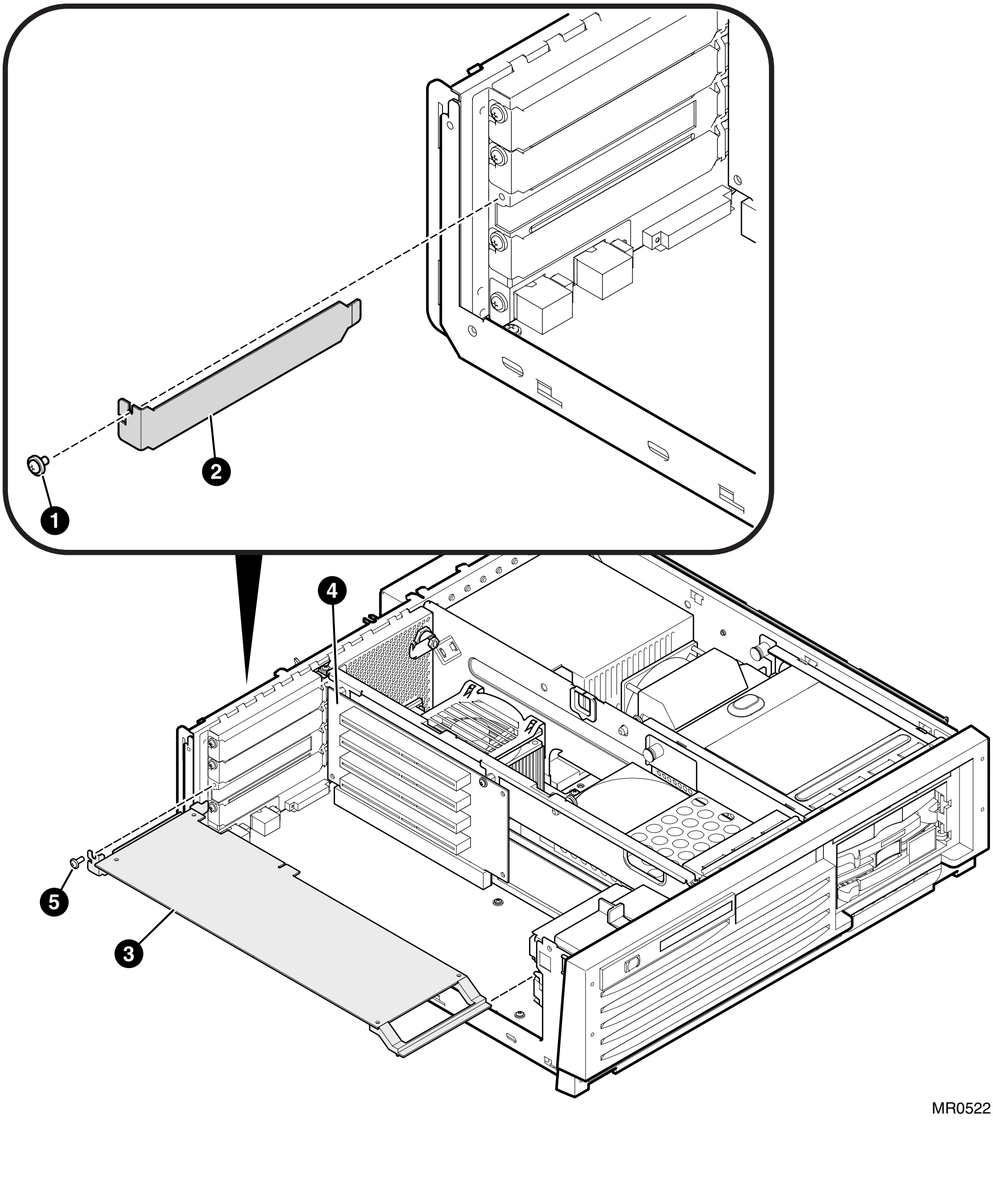
❸ Slot 3 – 33 MHz, 3.3v

❹ Slot 4 – 33 MHz, 3.3v

❺ LED – connected to +5 VAUX

When installing PCI option modules, you do not normally need to perform any configuration procedures. The system configures PCI modules automatically. But because some PCI option modules require configuration CDs, refer to the documentation for that PCI option module.

Figure 8–13 Replacing or Installing a PCI Option Module



Replacing or Installing a PCI Module

CAUTION: *Check the keying before you install the PCI option module and do not force it into place. Plugging a module into the wrong slot can damage it.*
5v cards are not allowed.

---

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the side panel as explained in Section 8.6.
4. Remove the slot cover screw ❶, slide out the slot cover ❷, and set it aside.
5. To remove a PCI option module ❸, grasp it at the corners and pull it straight out.
6. To install a PCI option module, grasp it at the corners and push it into the appropriate unused slot in the PCI riser card ❹.
7. Insert the retaining screw ❺ to secure the module.
8. Replace the side panel as explained in Section 8.6.
9. Replace the top cover as explained in Section 8.5.
10. Reconnect the power cord, turn on system power, and boot the system.

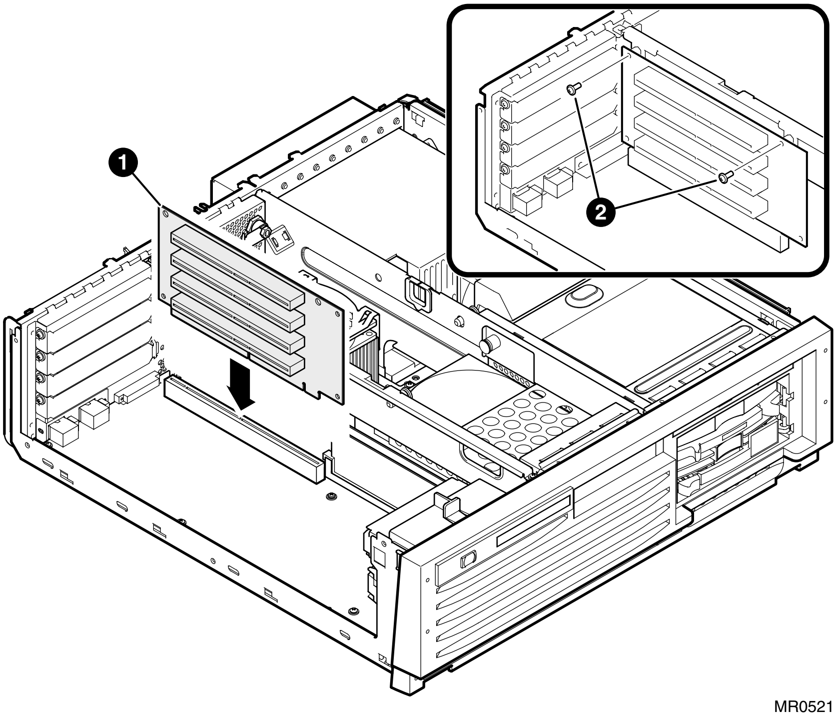
Verification

1. Turn on power to the system.
2. At the >>> prompt, enter the SRM **show config** command. Examine the PCI bus information in the display to make sure that the new option is listed.
3. If you installed a bootable device, enter the SRM **show device** command to determine the device name.

8.14 Replacing the PCI Riser Card

The PCI riser card provides slots for PCI options modules and connects them to the motherboard.

Figure 8–14 Replacing the PCI Riser Card



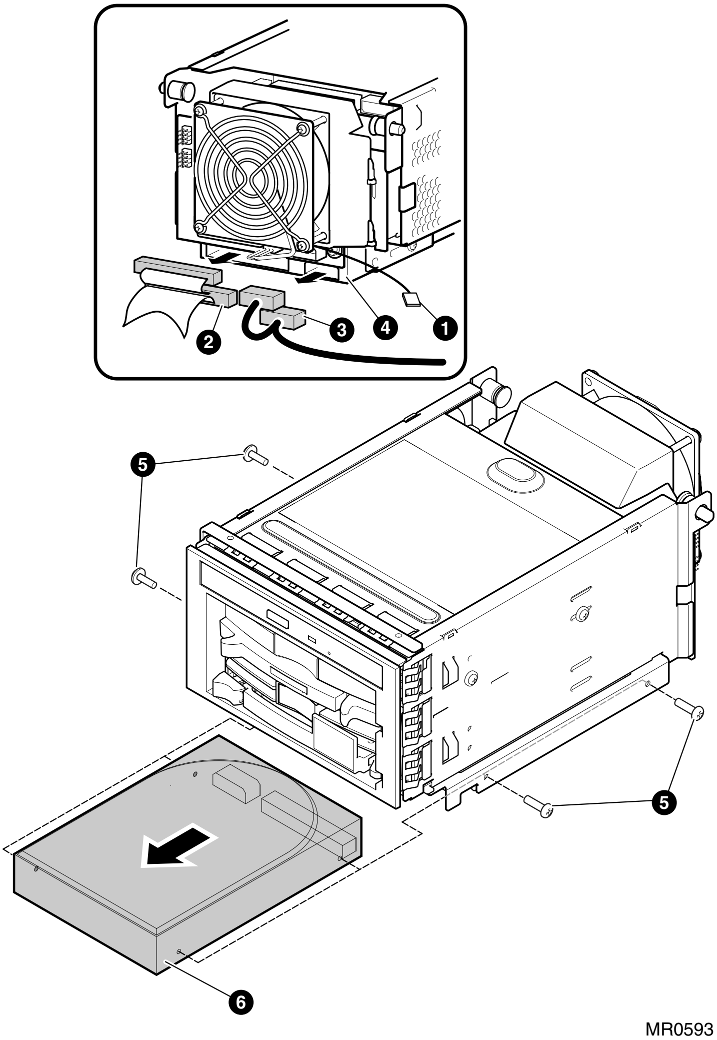
Replacing the PCI Riser

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the side panel as explained in Section 8.6.
4. Remove all PCI option modules as explained in Section 8.13.
5. Remove the two screws ❷ from the top corners of the PCI riser card.
6. Grasp the card ❶ by its upper corners and pull it out of its slot.
7. Grasp the new PCI riser card by its upper corners so that the option slots face the open side of the enclosure. Push it down into its slot on the motherboard.
8. Line up the holes in the riser's upper corners with the holes in the support bracket and insert the two screws.
9. Reinstall all PCI option modules as explained in Section 8.13.
10. Replace the side panel as explained in Section 8.6.
11. Replace the top cover as explained in Section 8.5.
12. Reconnect the power cord, turn on system power, and boot the system.

8.15 Replacing Bottom Drive – Front Access Storage Cage

The bottom disk drive provides optional storage. Refer to the following figure and procedure when replacing a bottom drive.

Figure 8–15 Replacing Bottom Drive – Front Access Storage Cage



Replacing Bottom Drive – Front Access Storage Cage

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the front access storage cage and all cables as explained in Section 8.11.
5. Remove the four screws ❺ from the sides of the bottom drive ❻ and slide the drive from the cage.
6. Slide the new drive into the cage and insert the four screws.
7. Reinstall the front access storage cage as explained in Section 8.11.
8. Replace the top cover as explained in Section 8.5.
9. Reconnect the power cord, turn on system power, and boot the system.

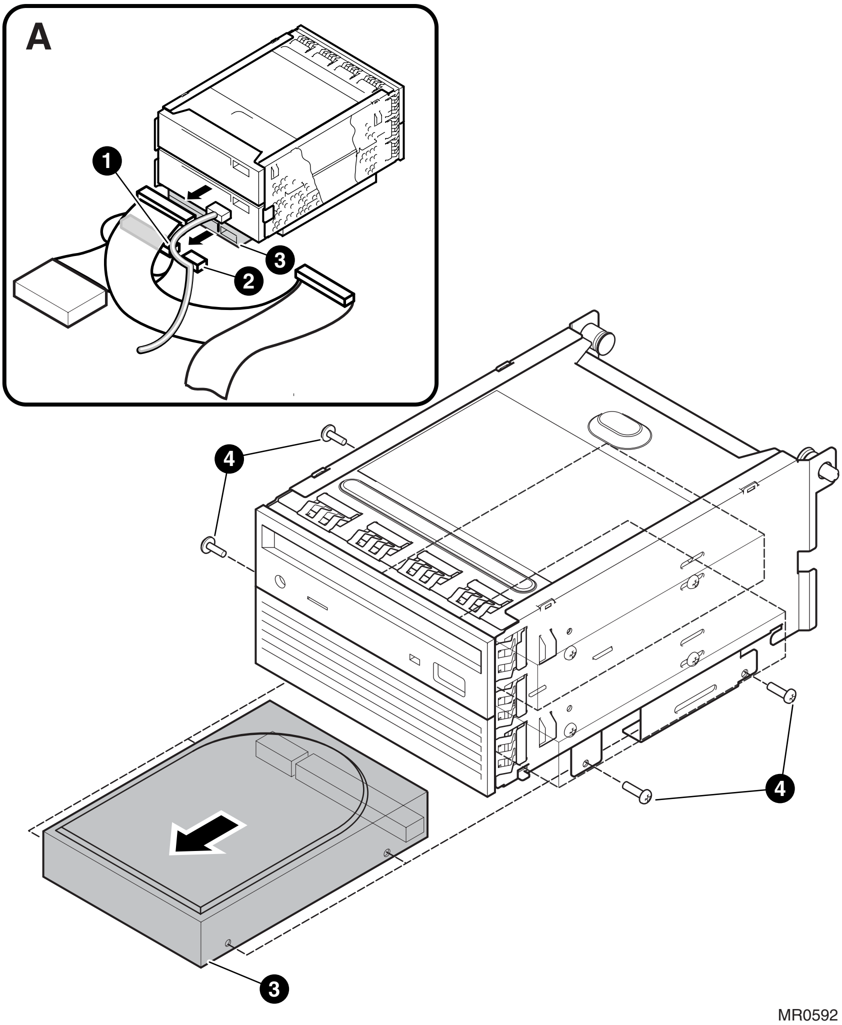
Verification

Enter the **init** command and use the **show device** command to verify that the system has identified the new drive.

8.16 Replacing Bottom Drive – Internal Storage Cage

The bottom disk drive provides optional storage. Refer to the following figure and procedure when replacing a bottom drive.

Figure 8–16 Replacing Bottom Drive – Internal Storage Cage



Replacing Bottom Drive – Internal Storage Cage

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the internal storage cage and all cables as explained in Section 8.12.
5. Remove the four screws ❹ from the sides of the bottom drive ❸ and slide the drive from the cage.
6. Slide the new drive into the cage and insert the four screws.
7. Reinstall the storage cage as explained in Section 8.12.
8. Replace the top cover as explained in Section 8.5.
9. Reconnect the power cord, turn on system power, and boot the system.

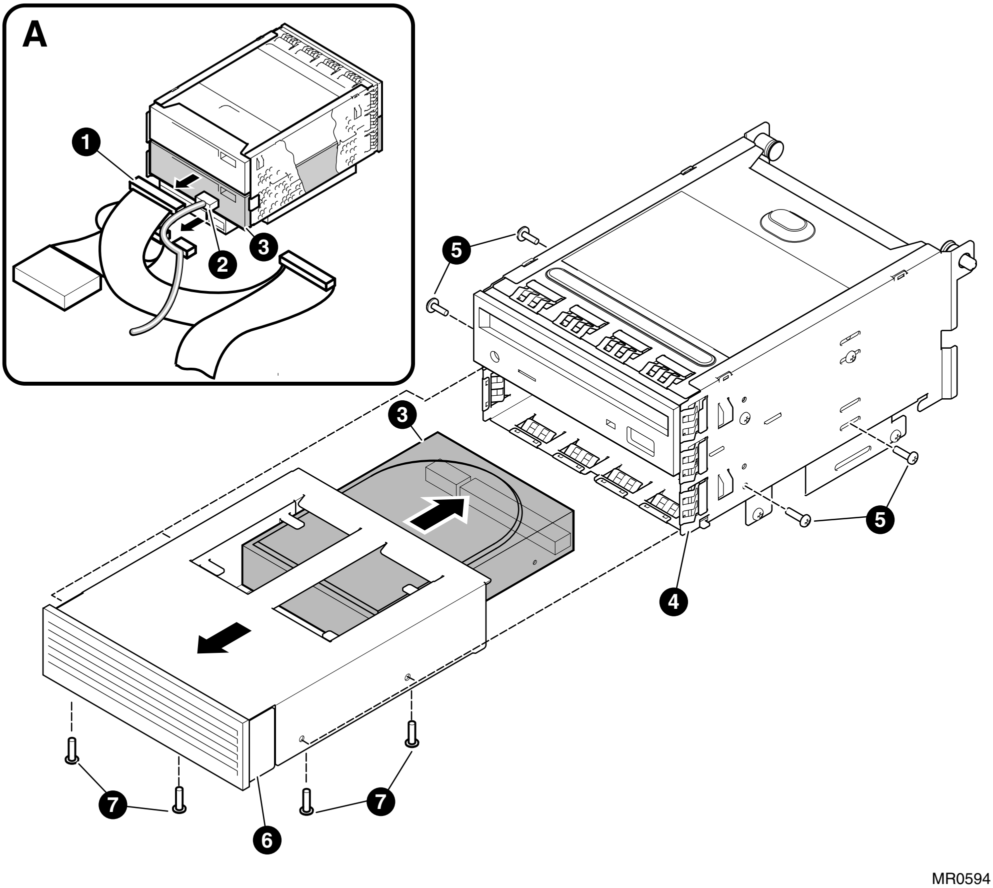
Verification

Enter the **init** command and use the **show device** command to verify that the system has identified the new drive.

8.17 Replacing Middle Drive – Internal Storage Cage

The middle drive provides optional storage. Refer to the following figure and procedure when replacing a middle drive.

Figure 8–17 Replacing Middle Drive – Internal Storage Cage



Replacing Middle Drive – Internal Storage Cage

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the internal storage cage ❹ and all cables as explained in Section 8.12.
5. Remove the four screws ❺ from the sides of the storage cage and slide the drive assembly ❻ from the storage cage.
6. Remove the four screws ❼ from the bottom of the drive assembly and slide the drive ❸ out. Set the drive aside.
7. Insert the new drive into the drive assembly and fasten the four bottom screws.
8. Slide the drive assembly into the storage cage and fasten the four side screws.
9. Reinstall the storage cage as explained in Section 8.12.
10. Replace the top cover as explained in Section 8.5.
11. Reconnect the power cord, turn on system power, and boot the system.

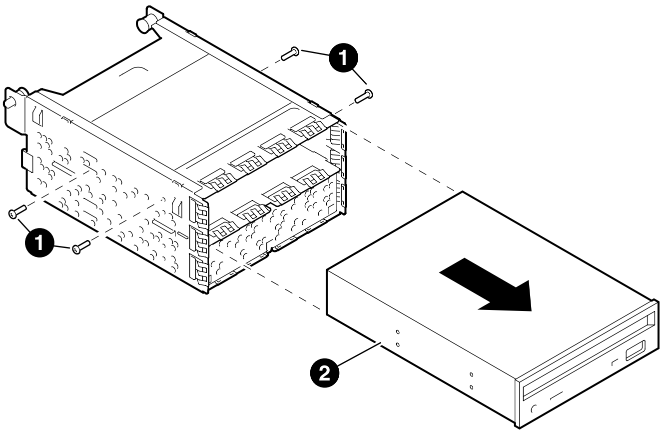
Verification

Enter the **init** command and use the **show device** command to verify that the system has identified the new drive.

8.18 Replacing DVD/CD-RW Drive – Internal Storage Cage

The top drive is a half-height DVD/CR-RW drive for use with optical media. Refer to the following figure and procedure when replacing a DVD/CD-RW drive.

Figure 8–18 Replacing DVD/CD-RW Drive – Internal Storage Cage



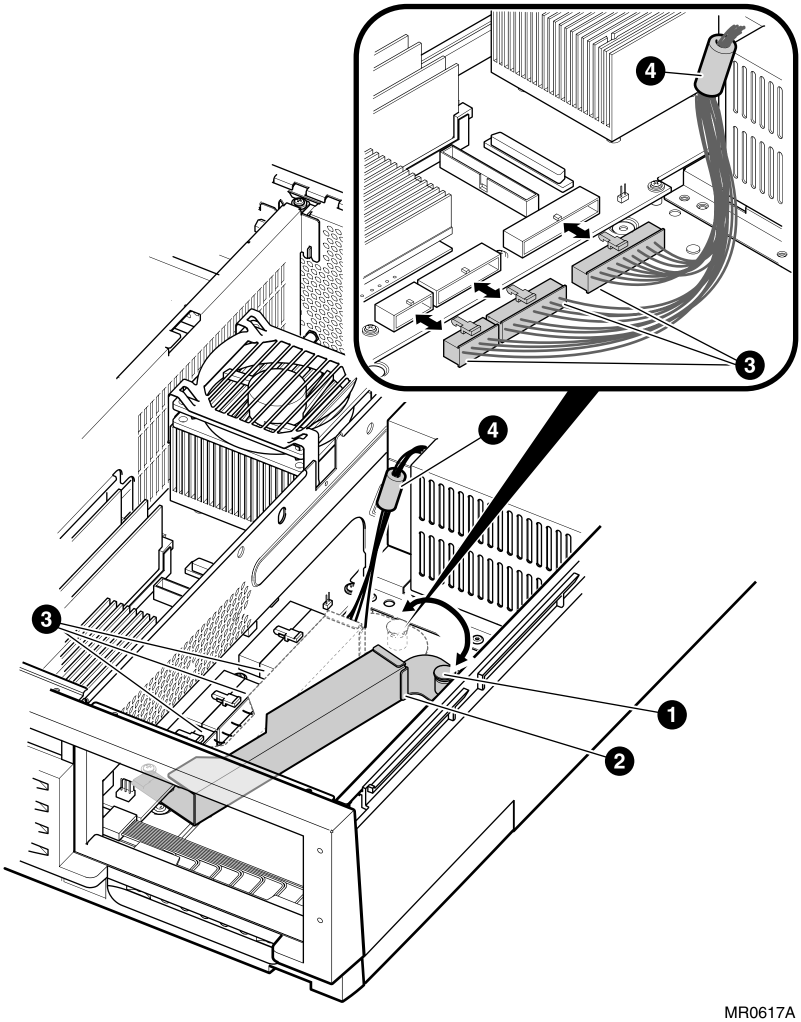
Replacing DVD/CD-RW Drive – Internal Storage Cage

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the internal storage cage and all cables as explained in Section 8.12.
5. Remove the four screws ➊ that fasten the drive ➋ to the storage cage.
6. Pull drive forward. Push away EMC finger stock clips if they hang up on the drive.
7. Before installing the new drive, be sure all eight (8) EMC finger stock clips are in place.
8. Slide the new drive into the storage cage and insert the four screws as shown.
9. Reinstall the storage cage as explained in Section 8.12.
10. Replace the top cover as explained in Section 8.5.
11. Reconnect the power cord, turn on system power, and boot the system.

8.19 Replacing the Power Supply

The power supply provides regulated power to the system. Refer to the following figure and procedure when replacing the power supply.

Figure 8–19 Removing Connectors from the Power Supply

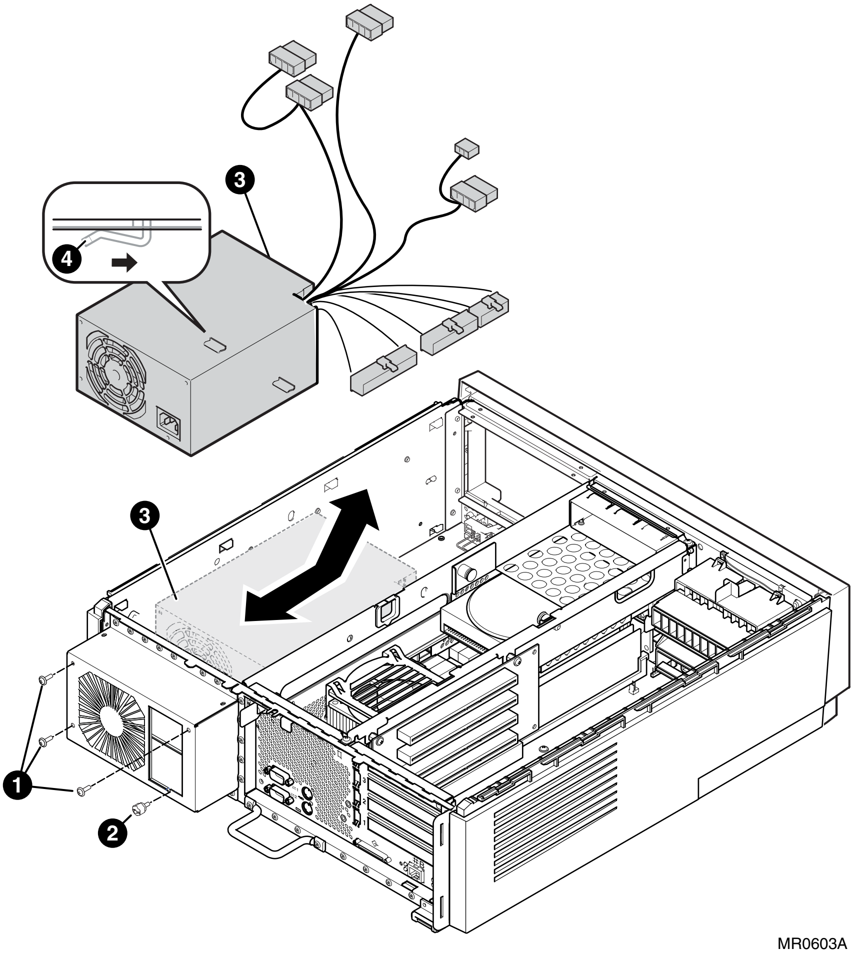


Removing Connectors from the Power Supply

WARNING: Hazardous voltages are contained within the power supply. Do not attempt to service. Return to factory for service.

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. To provide clearance for the power supply, remove the storage cage from that bay. Remove a front access storage cage and all cables as explained in Section 8.11. Remove an internal storage cage and all cables as explained in Section 8.12.
5. Lift the pull pin ❶ near the power supply and move that end of the cable channel ❷ to the side. The cable channel covers the cables running from the power supply to the motherboard.
6. Remove the three connectors ❸ from the motherboard by pressing the locking tab and pulling back on the connector. The connectors are different sizes, making it simple to reconnect.
7. Continue with the next procedure, Replacing the Power Supply.

Figure 8–20 Replacing the Power Supply



Replacing the Power Supply

1. Perform the steps as explained in the preceding procedure.
2. At the rear of the enclosure, remove the four screws from the power supply case. Three small screws ❶ require use of a screwdriver, while the fourth screw ❷ (thumbscrew) varies with the configuration. For rackmounted systems, a bracket is provided as a lock for the power cord.
3. Push the power supply ❸ into the enclosure until it stops. Lift up the cable end of the power supply, slide it out at a slight angle, and set it aside.
4. Before installing the new power supply, take note of two clips ❹ on its bottom. These clips must slide into matching channels in the enclosure.
5. Holding the power supply ❸ at a slight angle, slide it into the enclosure until it lays flat. Push the power supply into its corner until it stops. You should be able to feel the two bottom clips lock into place.
6. Insert the four screws ❶ ❷ into the power supply case.
7. Referring to Figure 8–19, plug the three connectors ❸ (from the power supply) into their sockets on the motherboard. Work from shortest to longest cable. Be sure to gather the cables together as tightly as possible so they fit under the cable channel ❷.
8. Slide the cable channel over the cables, carefully tucking the wires under the channel, and snap the pull pin ❶ into place.
9. Reinstall the storage cage into that bay. Install a front access storage cage and all cables as explained in Section 8.11. Install an internal storage cage and all cables as explained in Section 8.12.
10. Replace the top cover as explained in Section 8.5.
11. Reconnect the power cord, turn on system power, and boot the system.

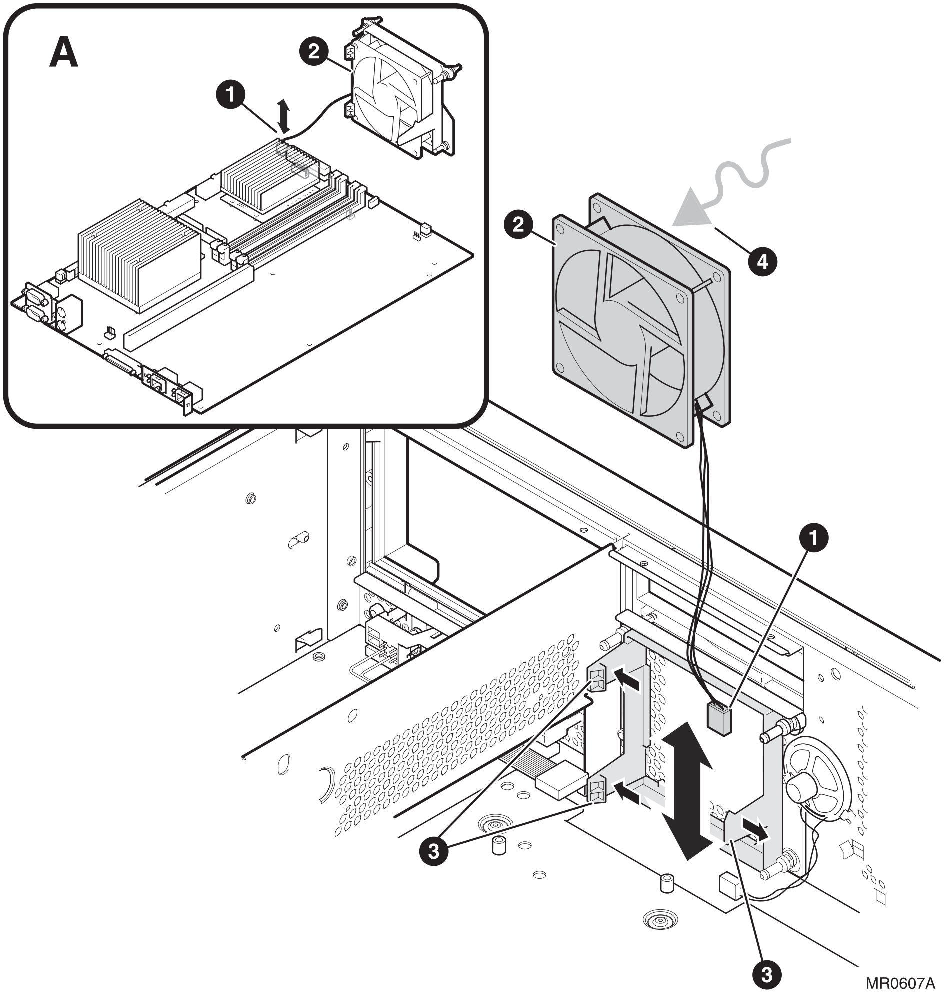
Verification

At the >>> prompt, enter the **show power** command or the RMC **status** command to verify system voltages.

8.20 Replacing the System Fan

The system fan, mounted in the center bay, provides additional cooling for the enclosure. Refer to the following figure and procedure when replacing the system fan.

Figure 8–21 Replacing the System Fan



WARNING: Contact with moving fan can cause severe injury to fingers. Avoid contact or remove power prior to access.

=== V @ >240VA

WARNING: High current area. Currents exceeding 240 VA can cause burns or eye injury. Avoid contact with parts or remove power prior to access.

Replacing the System Fan

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the center internal drive bay as explained in Section 8.9. This provides access to the fan connector.
5. Locate the fan ❷ at the front of the enclosure. Pull back the three clips ❸ that secure the fan and slowly work it free.
6. Unplug the fan connector ❶ from the motherboard and remove the fan.
7. Orient the new fan so that the airflow ❹ is as shown.
8. Install the new fan by pushing it into the three clips until they snap into place.
9. Slide the fan cable under the partition and insert the connector into the motherboard at connector J3 (callout ❶ in inset A).
10. Replace the center internal drive bay as explained in Section 8.9.
11. Replace the top cover as explained in Section 8.5.
12. Reconnect the power cord, turn on system power, and boot the system.

Verification

1. Invoke the remote management console.
2. Enter the **env** command to verify the fan status.

8.21 Removing or Installing a Memory DIMM

The system supports a total of 4 DIMMs, divided into two arrays of two slots each, and located on the motherboard. DIMMs within an array must be of the same size and speed. The system supports a maximum of 4 GB of memory.

Only the following DIMMs and DIMM options can be used in an *AlphaServer* DS15 system.

Density	DIMM	DIMM Option (2 DIMMs per set)
512 MB	20-01EBA-09	3X-MS315-EA
1 GB	20-00FBA-09	3X-MS315-FA
2 GB	20-00GBA-09	3X-MS315-GA

CAUTION: *Using different DIMMs may result in loss of data.*

Memory Configuration Rules

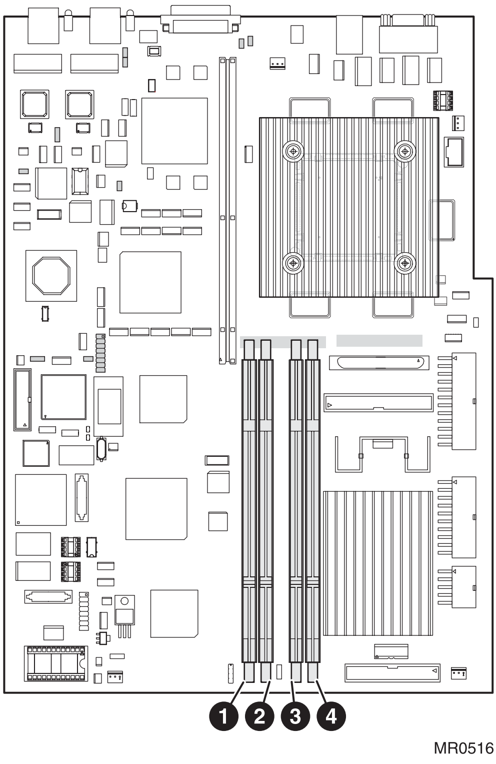
- You can install up to four (4) DIMMs on the motherboard.
- A maximum of 4 GB of memory is supported.
- There are two memory arrays, numbered 0 and 2, with two slots per array.
- A memory array must be populated with two DIMMs of the same size and speed. (See the table above for supported sizes and capacity.)
- Memory arrays must be populated in numerical order, starting with array 0.

Table 8–4 DIMM and Array Reference

	DIMM	Connector	Array
❶	1	J12	0
❷	3	J13	2
❸	0	J14	0
❹	2	J15	2

The DIMMs in the preceding table are located as shown in the following figure.

Figure 8–22 Locations for DIMMs on the Motherboard



WARNING: To prevent injury, access is limited to persons who have appropriate technical training and experience. Such persons are expected to understand the hazards of working within this equipment and take measures to minimize danger to themselves or others.

WARNING: Do not remove memory DIMMs until the green LED on the PCI riser card is off (approximately 20 seconds after a power-down).

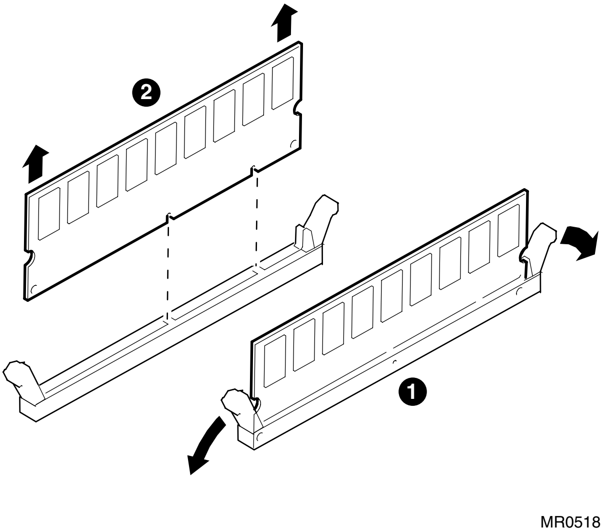
WARNING: Modules have parts that operate at high temperatures. Wait 2 minutes after power is removed before touching any module.

WARNING: To prevent personal injury or damage to any of the system modules, unplug the power cord from the power supply before installing components. Make sure the power LEDs are not lit before removing or replacing modules.

8.21.1 Removing a Memory DIMM

Memory DIMMs are critical components and should be handled with care. Refer to the following figure and procedure when replacing a memory DIMM.

Figure 8–23 Removing a Memory DIMMs



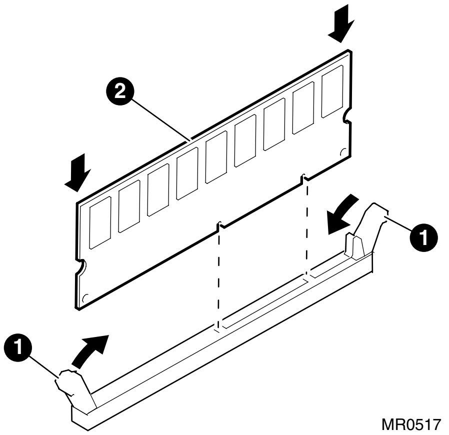
Removing a Memory DIMM

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the center internal storage bay as described in Section 8.9.
5. Use Table 8–4 and Figure 8–22 to determine the location of all DIMMs.
6. Release the clips ❶ securing the affected DIMM ❷, grasp it by its top corners, and pull upward. Note the capacity and slot location of the DIMM.
7. Continue with the next procedure, Installing a Memory DIMM.

8.21.2 Installing a Memory DIMM

Memory DIMMs must be replaced according to the rules in preceding Section 8.21. Refer to the following figure and procedure when replacing or installing a DIMM.

Figure 8–24 Installing a Memory DIMM



Installing a Memory DIMM

1. Perform the steps in the preceding procedure, Removing a Memory DIMM.
2. Use preceding Table 8–4, Figure 8–22, and the memory configuration rules to determine the proper location in which to install the new memory DIMM.
3. Make sure the clips ❶ are pushed down and away from the memory socket.
4. Remove the DIMM ❷ from the static-free-container and hold it by its left and right edges.
5. Align the notches on the DIMM's gold fingers with the connector keys in the memory slot, and push the DIMM down firmly into the slot until the clips ❶ snap into place. Verify that the clips are engaged.
6. Replace the center internal drive bay as explained in Section 8.9.
7. Replace the top cover as explained in Section 8.5.
8. Reconnect the power cord, turn on system power, and boot the system.

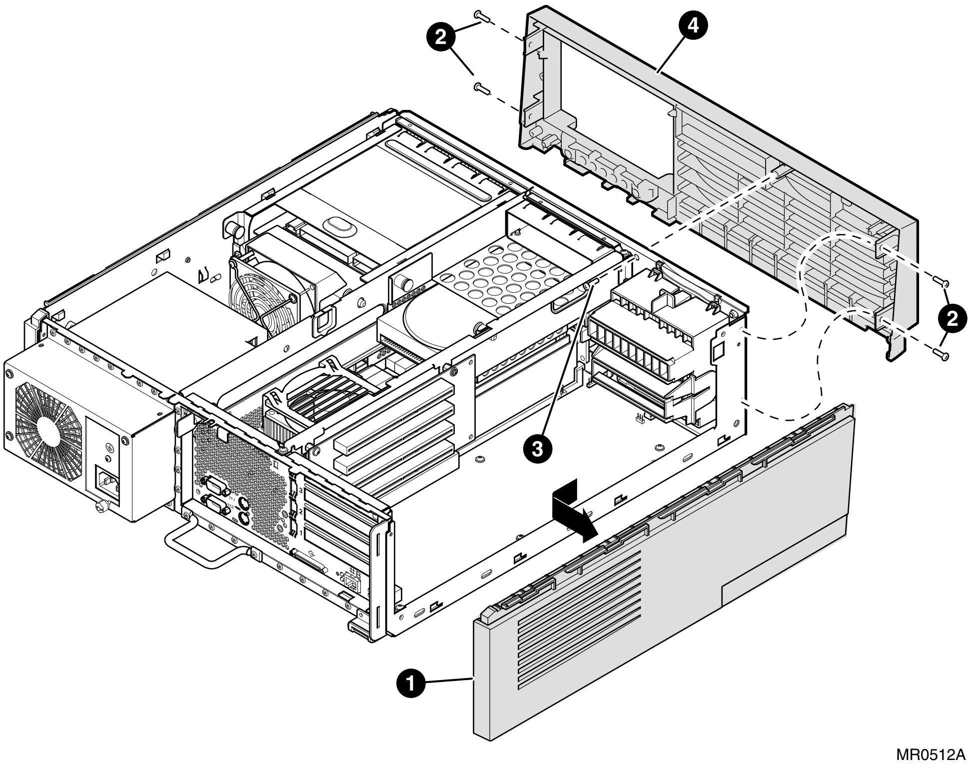
Verification

1. At the SRM console prompt, issue the **buildfru -dimm** command to provide each new DIMM with a unique serial number.
2. Issue the **show memory** command to display the amount of memory in each array and the total amount of memory in the system.

8.22 Replacing the Operator Control Panel

The operator control panel (OCP) provides system controls and status indicators. The OCP is accessible after removing the top cover, side panel, and front bezel. Refer to the following figures and procedure when replacing an operator control panel.

Figure 8–25 Removing the Front Bezel



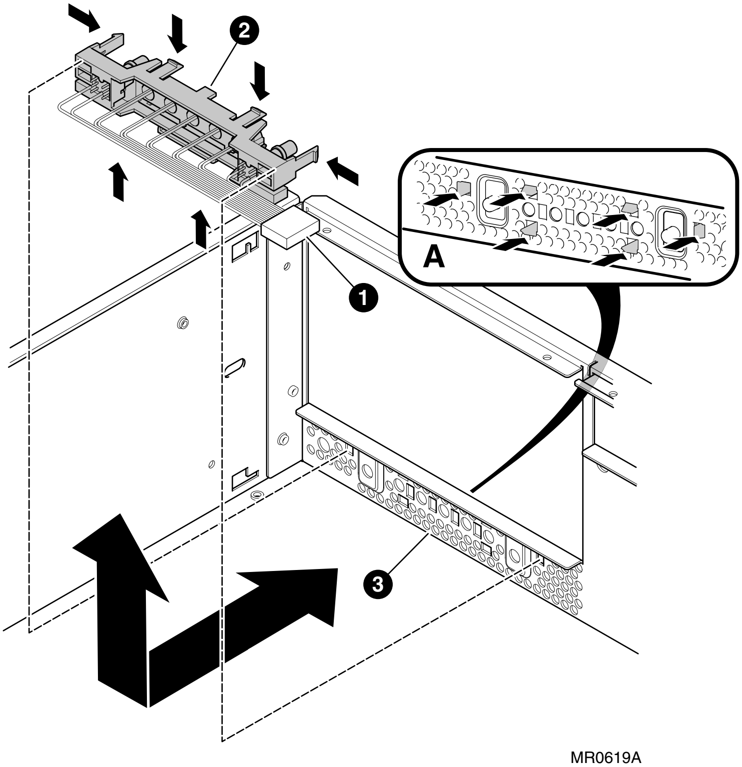
Removing the Front Bezel

CAUTION: *Care must be taken when installing a new OCP so that the LEDs line up with the holes in the enclosure. Failure to align the LEDs correctly may result in damage to an LED.*

---

1. Shut down the operating system.
3. Turn off system power and unplug the power cord from the power supply.
4. Remove the top cover as explained in Section 8.5.
5. Remove the side panel ❶ as explained in Section 8.6.
6. Remove the center internal storage bay as explained in Section 8.9.
7. If installed, remove the front access storage cage and all cables as explained in Section 8.11. Otherwise, remove the internal storage cage and all cables as explained in Section 8.12.
8. Remove the front bezel. To do this, remove the four side screws ❷ and one front screw ❸, then remove the front bezel ❹.

Figure 8–26 Replacing the Operator Control Panel



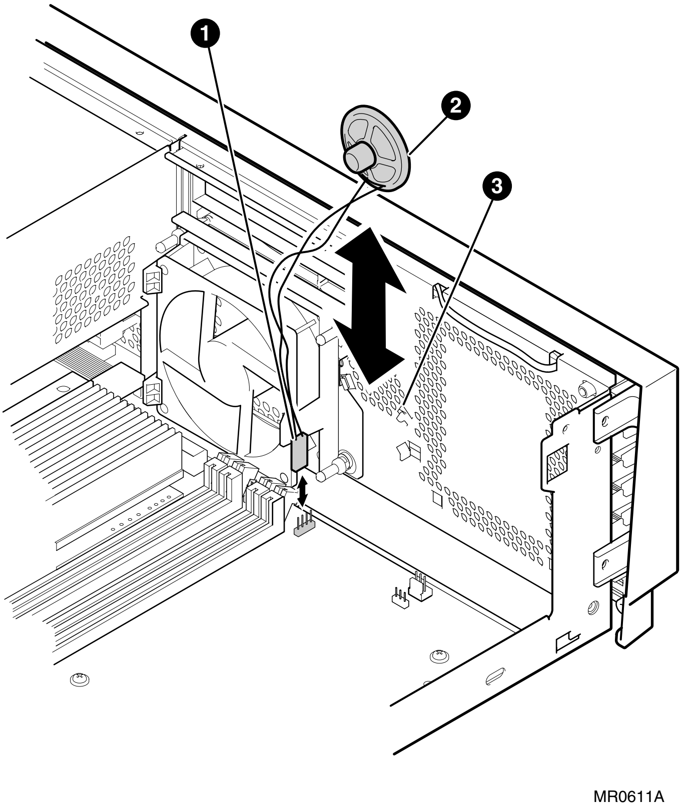
Replacing the Operator Control Panel

1. After removing the front bezel (as explained in the preceding procedure), unplug the OCP connector ❶ from the motherboard.
2. Push in the tabs (shown in insert A) that fasten the OCP ❷ to the front panel ❸.
3. Pull the OCP away from the front panel and remove the two button caps.
4. Put the button caps on the new OCP, and snap the new OCP back into place inside the front panel. Be sure to align the LEDs with their mounting holes in the enclosure.
5. Plug the OCP connector into connector J7 on the motherboard.
6. Reinstall the front bezel by inserting the four side screws and one front screw.
7. Reinstall the side panel as explained in Section 8.6.
8. Reinstall either the front access storage cage (as explained in Section 8.11) or the internal storage cage (as explained in Section 8.12).
9. Reinstall the center internal storage bay as explained in Section 8.9.
10. Replace the top cover as explained in Section 8.5.
11. Reconnect the power cord, turn on system power, and boot the system.

8.23 Replacing the Speaker

The speaker provides audible tones for various system events. Refer to the following figure and procedure when replacing a speaker.

Figure 8–27 Replacing the Speaker



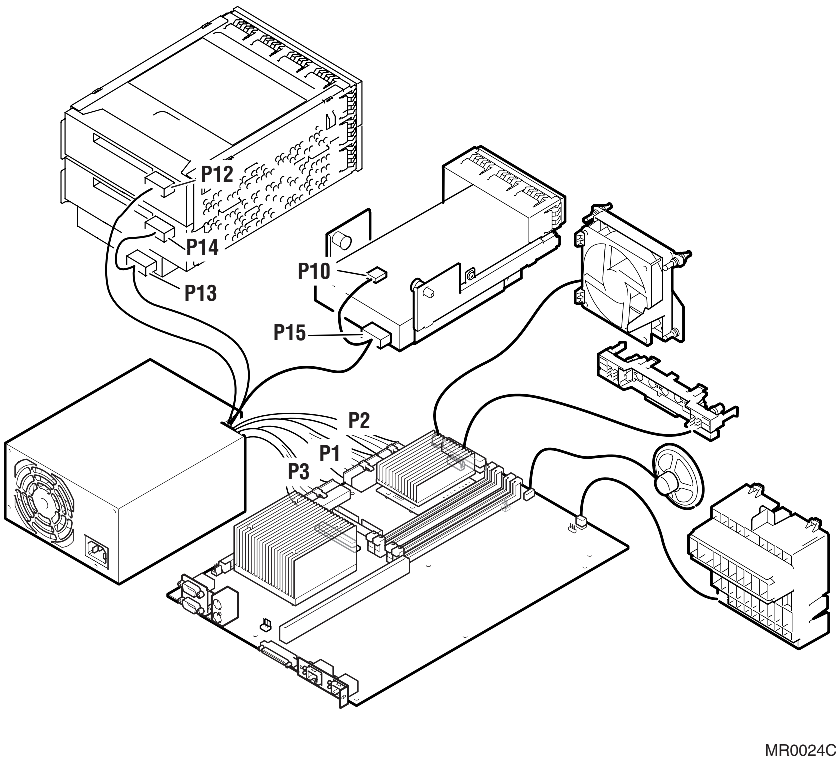
Replacing the Speaker

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove the top cover as explained in Section 8.5.
4. Remove the center internal drive bay as explained in Section 8.9.
5. Remove the speaker connector ➊ from the motherboard.
6. Slide the speaker ➋ upward from its retaining clips ➌ and set it aside.
7. Slide the new speaker down and into its retaining clips.
8. Insert the speaker connector ➊ into connector J2 on the motherboard. To properly connect the fan, note that pin 1 is marked on the motherboard and the red speaker wire with a small black dot connects to pin 1.
9. Reinstall the center internal drive bay as explained in Section 8.9.
10. Replace the top cover as explained in Section 8.5.
11. Reconnect the power cord, turn on system power, and boot the system.

8.24 Preparing to Replace the Motherboard

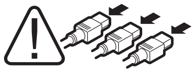
The motherboard is the main logic board for the system and is mounted on the bottom of the enclosure. You must remove virtually all components to access the motherboard. The following figure (also available on the inside of the enclosure cover) depicts most of the components that need to be removed.

Figure 8–28 Components Connected to the Motherboard

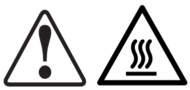


8.25 Removing Intervening Components

To access the motherboard, you must remove most of the other system components. Refer to the following figure and procedure when removing the intervening components.



WARNING: To prevent personal injury or damage to any of the system modules, unplug the power cord from the power supply before touching components. Make sure the power LEDs are not lit before removing or replacing modules.



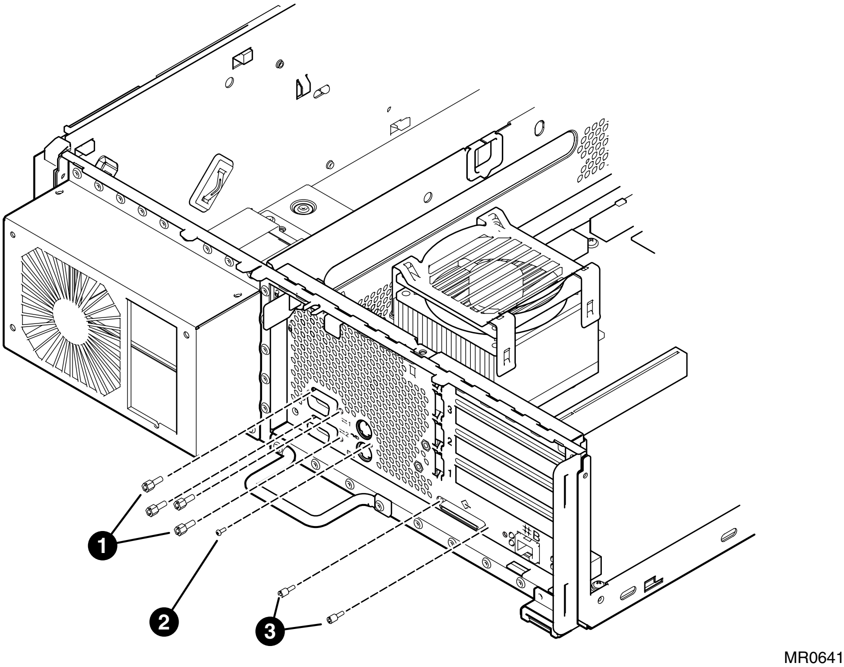
WARNING: Modules have parts that operate at high temperatures. Wait 2 minutes after power is removed before touching any module.

CAUTION: *When removing the system motherboard, be careful not to flex the board. This can result in damage to the circuitry.*

NOTE: *Replacing the system motherboard requires the removal of other FRUs. Review the removal procedures for the fans, CPUs, and options before beginning the system motherboard removal procedure. Mark the original locations of all components and cables as you remove them.*

A cordless screwdriver is highly recommended for these procedures.

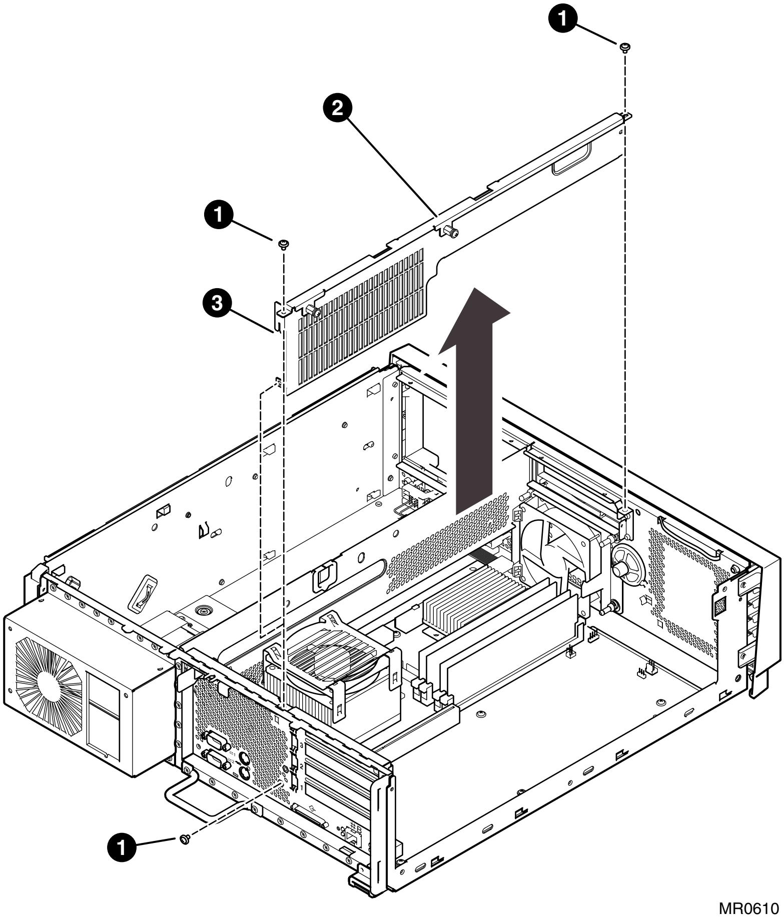
Figure 8–29 Removing Rear Screws



Removing Intervening Components

1. Shut down the operating system.
2. Turn off system power and unplug the power cord from the power supply.
3. Remove all external cables from the rear of the enclosure.
4. Remove the four hex screws ❶ that fasten the COM ports, the one mouse/keyboard screw ❷, and one screw ❸ from the SCSI connector, as shown in the preceding figure.
5. Remove the top cover as explained in Section 8.5.
6. Remove the side panel as explained in Section 8.6.
7. Remove all PCI option modules as explained in Section 8.13.
8. Remove the PCI riser card as explained in Section 8.14.
9. Remove the PCI fan as explained in Section 8.7.
10. Remove the center internal storage bay as explained in Section 8.9.
11. Remove the center support bracket as shown on the following page.

Figure 8–30 Removing the Center Support Bracket



Removing the Center Support Bracket

1. Perform the steps in the preceding procedure, Removing Intervening Components.
2. Remove the three retaining screws ❶ from the top and rear side of the bracket ❷. Lift the bracket up and set it aside.

Removing the Remaining Components

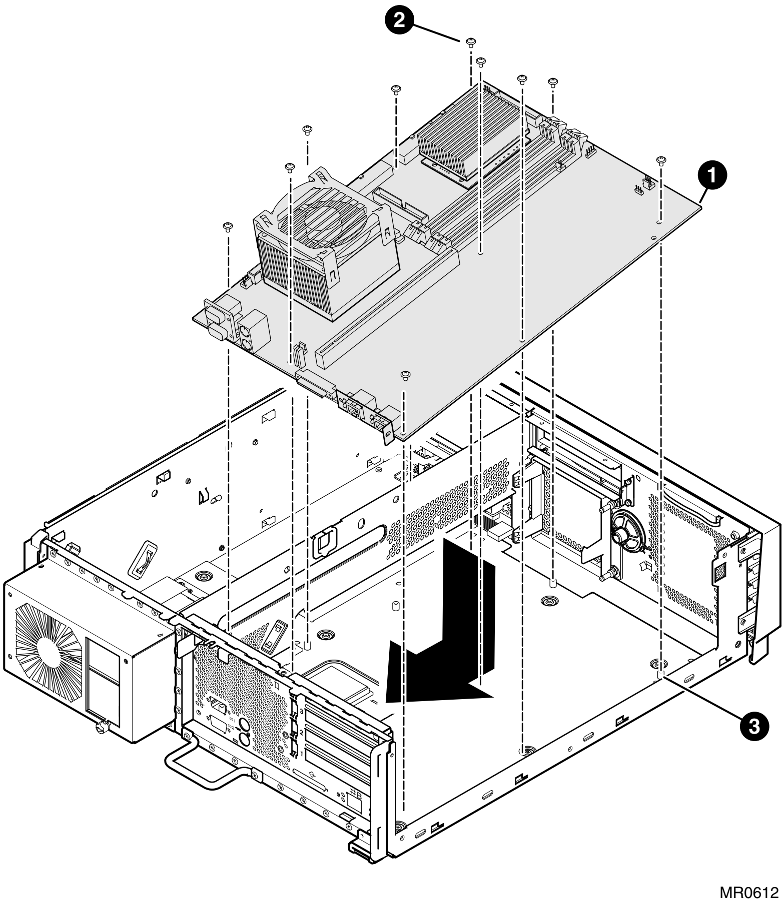
1. If installed, remove the front access storage cage and all cables as explained in Section 8.11. Otherwise, remove the internal storage cage and all cables as explained in Section 8.12.
2. Remove the system fan as explained in Section 8.20.
3. Remove the connector for the operator control panel as explained in Section 8.22.
4. Remove the power supply cables that are under the channel as explained in Section 8.19.

You are now ready to remove and replace the motherboard.

8.26 Replacing the Motherboard

The motherboard requires care in handling. Pay close attention to nearby metal brackets and sharp edges when replacing this component. Refer to the following figure and procedure when replacing the motherboard.

Figure 8–31 Replacing the Motherboard



Removing the Motherboard

1. After removing all intervening components as described in the preceding procedures, release the motherboard ❶ by removing all the screws ❷ that securing the motherboard to the enclosure.
2. Slide the motherboard slightly away from the rear face of the enclosure.
3. Lift the edge of the motherboard near the front of the enclosure and carefully lift it out. Set it aside.
4. When removing the motherboard, be careful not to lose the small metal shield on the SCSI connector. Remove the shield and set it aside because you will need to install it on the new motherboard.

Installing the New Motherboard

1. Slide the metal shield removed from the old motherboard onto the SCSI connector. Be careful not to drop the shield when installing motherboard into the enclosure.
2. Hold the motherboard ❶ with its COM port end pointing down toward the rear of the enclosure.
3. Slide the motherboard gently toward the rear of the enclosure until the I/O connectors on the motherboard pass through the openings on the rear panel of the enclosure.
4. Insert the four hex screws ❶ that fasten the COM ports, the one mouse/keyboard screw ❷, and one screw ❸ from the SCSI connector, as shown in Figure 8–29.
5. Insert all the retaining screws ❷ through the motherboard into the bottom of the enclosure.
6. Note the positions of all jumpers on the old motherboard and be sure to correctly set the jumpers on the new motherboard. See Appendix A for more information.

8.27 Reinstalling System Components

After installing the new motherboard, you need to replace all the other components and cables. Refer to the following procedure when reinstalling the system components.

Reinstalling System Components

1. Reinstall the power supply as explained in Section 8.19.
2. Plug the OCP connector into connector J7 on the motherboard as explained in Section 8.22.
3. Reinstall the system fan as explained in Section 8.20.
4. Reinstall the storage cage into that bay. Install a front access storage cage and all cables as explained in Section 8.11. Install an internal storage cage and all cables as explained in Section 8.12.
5. Reinstall the center support bracket as explained in Section 8.25, Figure 8–29.
6. Reinstall the center internal storage bay as explained in Section 8.9.
7. Reinstall the PCI fan as explained in Section 8.7.
8. Reinstall the PCI riser card as explained in Section 8.14.
9. Reinstall all PCI option modules as explained in Section 8.13.
10. Reinstall the side panel as explained in Section 8.6.
11. Replace the top cover as explained in Section 8.5.
12. Reconnect the power cord, turn on system power, and boot the system.

After Installing a New Motherboard:

1. Power up to the P00>>> prompt.
2. Enter the **clear_error all** command.
3. Enter the **set sys_serial_num** command to set the system serial number. (The serial number is on a label on the back of the system.) For example:

```
>>> set sys_serial_num NI900100022
```

IMPORTANT: *The system serial number must be set correctly. System Event Analyzer will not work with an incorrect serial number.*

The serial number propagates to all FRU devices that have EEPROMs.

Appendix A
Jumpers on System Motherboard

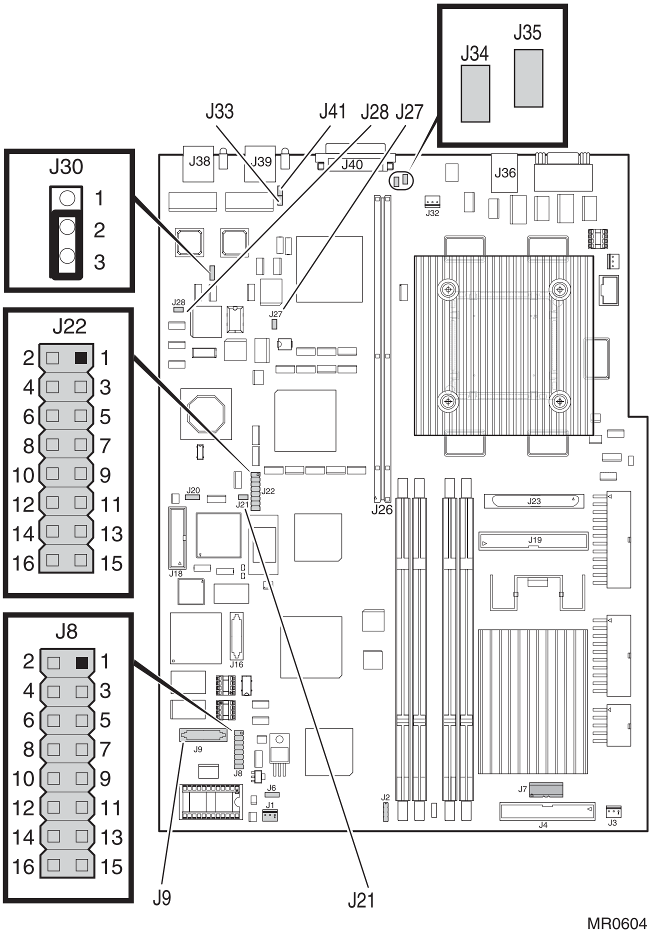
This appendix describes the configuration of jumpers on the system motherboard. Sections are as follows:

- Locations of Jumpers
- Function of Jumpers
- Setting Jumpers

A.1 Location of Jumpers

The following figure shows the location of all jumpers on the system motherboard. The next section describes the function and pin numbers for each jumper location. Several jumpers are supplied with the board at jumper J8 and other locations.

Figure A–1 Locations of Jumpers



A.2 Function of Jumpers

Jumpers can be grouped into system-level functions and server management functions.

A.2.1 System Jumpers

System jumpers are used for system-level functions. The default state for each jumper is off, that is, no jumper is installed. Refer to preceding Figure A–1 for the locations of these jumpers.

Table A–1 Jumpers for System-Level Functions

Jumper	Name	Pins	Function (Off = No Jumper)
J8	Failsafe loader	1 – 2	Off = Load SRM from flash ROM On = Load Failsafe image from flash ROM
J8	Mini-debugger	3 – 4	Off = Continue to boot On = Jump to mini-debugger
J8	Reserved	5 – 6	
J8	Reserved	7 – 8	
J8	Failsafe Flash Update	9 – 10	Off = Enable Failsafe flash updates On = Disable Failsafe flash updates
J8	Console Flash Update	11 – 12	Off = Enable Console flash updates On = Disable Console flash updates
J8	Tig Load	13 – 14	Off = Load from flash On = Load from PROM
J8	Floppy Load	15 – 16	Off = Load from flash On = Load from diskette
J21	Zircon Flash Update		Off = Enable Zircon flash updates On = Disable Zircon flash updates
J22	SROM Load	1 - 2	Off = Continue loading from flash On = Load from serial device only
J27	SCSI Port A Termination Power		Off = Termination enabled with firmware On = Onboard termination disabled
J33	SCSI Port B Termination Power		Off = Termination enabled with firmware On = Onboard termination disabled
J34	SCSI bus width		Off = 8-bit SCSI on Channel A On = Wide 16-bit SCSI on Channel A

Jumper	Name	Pins	Function (Off = No Jumper)
	Channel A		(default)
J35	SCSI bus width Channel B		Off = 8-bit SCSI on Channel B On = Wide 16-bit SCSI on Channel B (default)
J41	SCSI Port B Terminator		Off = Enable SCSI terminator On = Disable SCSI terminator and enable shared bus (Tru64 UNIX)

A.2.2 Server Management Jumpers

Server management jumpers control functions related to the Server Management (SM) subsystem. The "Feature" jumpers provide an extended set of features to the Remote Management Console (RMC). The other jumpers either configure portions of the SM logic or provide information to Zircon. The following table explains the function of these jumpers.

The default state for these jumpers is normally off (no jumpers). Refer to preceding Figure A–1 for the locations of these jumpers.

Table A–2 Server Management Jumpers

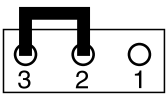
Jumper	Name	Pins	Function (Off = No Jumper)
J22	Feature_4	3 – 4	Off = Disk cage fan is monitored On = Disk cage fan is not monitored
J22	Feature_3	5 – 6	Off = Power-on only with riser installed On = Power-on without riser installed
J22	RMC Booter Write Enable	7 – 8	Off = RMC booter updates disabled On = RMC booter updates enabled
J22	Restore	9 – 10	Off = Normal operation On = Restore default values for environmental settings
J22	Feature_2	11 – 12	Off = Normal operation On = Forces booter into update mode
J22	Halt/Reset	13 – 14	Off = Halt when button is pressed On = Reset when button is pressed
J22	Feature0	15 – 16	Off = EEPROM checks enforced On = EEPROM checks not enforced

Jumper	Name	Pins	Function (Off = No Jumper)
J28	DTR		Off = Normal On = Force DTR

A.2.3 Jumper for COM1 Pass through Enable

Jumper J30 enables or disables the COM1 pass through mode. The settings are show in the following figure. Unlike other jumper settings, normal mode requires a jumper to be installed.

Table A–3 Jumper to Enable COM1 Pass through Mode

	Jumper	Function
J30 	1 – 2	Always pass through the RMC (default)
	2 – 3	Normal operation, user selects modes with COM1_MODE environment variable
	None	Always bypass the RMC

A.3 Setting Jumpers

Review the material in the previous sections of this appendix before setting any system jumpers. First, shut down the system and remove the power cord from the power supply.

CAUTION: *Static electricity can damage integrated circuits. Always use a grounded wrist strap (29-26246) and grounded work surface when working with internal parts of a computer system.*

Remove jewelry before working on internal parts of the system.

Setting Jumpers

1. Shut down the operating system.
2. Shut down power on all external options connected to the system.
3. Turn off power to the system.
4. Unplug the power cord from each power supply and wait for all LEDs to turn off.
5. Remove the top cover (as explained in Chapter 8) to gain access to the system motherboard.
6. If you need to remove a Field Replaceable Unit (FRU) to set jumpers, see Chapter 8.
7. Locate the jumper you need to set. Refer to Figure A–1 in this appendix. Set the jumpers as needed.
8. Reinstall any FRUs you removed.
9. Reinstall the enclosure panels.
10. Plug the power cords into the supplies.

Appendix B
Isolating Failing DIMMs

This appendix explains how to manually isolate a failing DIMM from the failing address and failing data bits. It also covers how to isolate single-bit errors. The following topics are covered:

- Information for Isolating Failures
- DIMM Isolation Procedure
- EV68 Single-Bit Errors

B.1 Information for Isolating Failures

Table B–1 lists the information needed to isolate the failure. The failing address and failing data can come from a variety of different locations such as the SROM serial line, SRM screen displays, the SRM event log, and errors detected by the 21264 (EV68) chip.

Convert the address to data bits if the address is not on a 256-bit alignment (address ends in a value less than 20 or address *xxxxx*20 or address *xxxxxnn*, where *nn* is 1 through 1F). For example, using failing address 0x1004 and failing data bit 8(dec), first multiply the failing address 4 by 8 = 32. Then add 32 to the failing data bit to yield the actual failing data bit 40. This conversion yields the new failing information to be failing address 0x1000 and failing data bit = 40(dec).

Table B–1 Information Needed to Isolate Failing DIMMs

Failing Address	
Failing Data/Check bits	
Array Address Registers (AARs)	**Memory Addresses**
CSC	801.A000.0000
AAR0	801.A000.0100
AAR2	801.A000.0180
DPR Locations	**Memory Addresses**
DPR:80	801.1000.2000
DPR:84	801.1000.2100

NOTE: *Arrays 1 and 3 do not exist on the AlphaServer DS15. Registers for these arrays (AAR1, AAR3, DPR 82, and DPR 86) are always zero.*

B.2 DIMM Isolation Procedure

Use the following procedure to isolate a failing DIMM.

1. Find the failing array by using the failing address and the Array Address Registers. Use the AAR base address and size to create an Address range for comparing the failing address.
2. Determine if the Address XORing is enabled.
   - If Address XORING is enabled, use Table B–2 to find the real array on which two-way interleaving has failed.
   - If Bit 51 of the CSC register is set to 1, XORing is disabled.

Table B–2 Determining the Real Failed Array for 2-Way Interleaving

Failing Address <8>	Original Array 0	Original Array 2
0	Real Array 0	Real Array 2
1	Real Array 2	Real Array 0

3. After finding the real array, determine whether it is the lower array set or the upper array set. Use DPR locations 80 and 84 listed in Table B–1. Table B–3 shows the description of these locations.

Table B–3 Description of DPR Locations 80, and 84

DPR Location	Description	
80	Array 0 (AAR 0) Configuration	
	Bits<7:4>	Bits<3:0>
	4 = non split—lower set only	0 = Configured—Lowest array
		1 = Configured—Next lowest array
	5 = split—lower set only	4 = Misconfigured—Missing DIMM(s)
		8 = Misconfigured—Illegal DIMM(s)
		C = Misconfigured—Incompatible DIMM(s)
84	Array 2 (AAR 2) configuration	

4. Now that you have the real array, the failing Data/Check bits, and the correct set, use Table B–4 to find the failing DIMM or DIMMs.

The table shows data bits 0–127 and check bits 0–15. These data bits indicate a single-bit error. An SROM compare error would yield address and data bits from 0–63. When you convert the address to be in the correct range, the failing data would be somewhere between 0 and 127.

Table B–4 Failing DIMM Lookup Table

	Array 0		Array 2	
Data Bits	DIMM	J#	DIMM	J#
0	0	14	2	15
1	0	14	2	15
2	0	14	2	15
3	0	14	2	15
4	0	14	2	15
5	0	14	2	15
6	0	14	2	15
7	0	14	2	15
8	0	14	2	15
9	0	14	2	15
10	0	14	2	15
11	0	14	2	15
12	0	14	2	15
13	0	14	2	15
14	0	14	2	15
15	0	14	2	15
16	1	12	3	13
17	1	12	3	13
18	1	12	3	13
19	1	12	3	13
20	1	12	3	13
21	1	12	3	13
22	1	12	3	13
23	1	12	3	13

Table B–4 Failing DIMM Lookup Table (Continued)

	Array 0		Array 2	
Data Bits	DIMM	J#	DIMM	J#
24	1	12	3	13
25	1	12	3	13
26	1	12	3	13
27	1	12	3	13
28	1	12	3	13
29	1	12	3	13
30	1	12	3	13
31	1	12	3	13
32	0	14	2	15
33	0	14	2	15
34	0	14	2	15
35	0	14	2	15
36	0	14	2	15
37	0	14	2	15
38	0	14	2	15
39	0	14	2	15
40	0	14	2	15
41	0	14	2	15
42	0	14	2	15
43	0	14	2	15
44	0	14	2	15
45	0	14	2	15
46	0	14	2	15
47	0	14	2	15

Table B–4 Failing DIMM Lookup Table (Continued)

	Array 0		Array 2	
Data Bits	DIMM	J#	DIMM	J#
48	1	12	3	13
49	1	12	3	13
50	1	12	3	13
51	1	12	3	13
52	1	12	3	13
53	1	12	3	13
54	1	12	3	13
55	1	12	3	13
56	1	12	3	13
57	1	12	3	13
58	1	12	3	13
59	1	12	3	13
60	1	12	3	13
61	1	12	3	13
62	1	12	3	13
63	1	12	3	13
64	0	14	2	15
65	0	14	2	15
66	0	14	2	15
67	0	14	2	15
68	0	14	2	15
69	0	14	2	15
70	0	14	2	15
71	0	14	2	15

Table B–4 Failing DIMM Lookup Table (Continued)

	Array 0		Array 2	
Data Bits	DIMM	J#	DIMM	J#
72	0	14	2	15
73	0	14	2	15
74	0	14	2	15
75	0	14	2	15
76	0	14	2	15
77	0	14	2	15
78	0	14	2	15
79	0	14	2	15
80	1	12	3	13
81	1	12	3	13
82	1	12	3	13
83	1	12	3	13
84	1	12	3	13
85	1	12	3	13
86	1	12	3	13
87	1	12	3	13
88	1	12	3	13
89	1	12	3	13
90	1	12	3	13
91	1	12	3	13
92	1	12	3	13
93	1	12	3	13
94	1	12	3	13
95	1	12	3	13

Table B–4 Failing DIMM Lookup Table (Continued)

	Array 0		Array 2	
Data Bits	DIMM	J#	DIMM	J#
96	0	14	2	15
97	0	14	2	15
98	0	14	2	15
99	0	14	2	15
100	0	14	2	15
101	0	14	2	15
102	0	14	2	15
103	0	14	2	15
104	0	14	2	15
105	0	14	2	15
106	0	14	2	15
107	0	14	2	15
108	0	14	2	15
109	0	14	2	15
110	0	14	2	15
111	0	14	2	15
112	1	12	3	13
113	1	12	3	13
114	1	12	3	13
115	1	12	3	13
116	1	12	3	13
117	1	12	3	13
118	1	12	3	13
119	1	12	3	13

Table B–4 Failing DIMM Lookup Table (Continued)

	Array 0		Array 2	
Data Bits	DIMM	J#	DIMM	J#
120	1	12	3	13
121	1	12	3	13
122	1	12	3	13
123	1	12	3	13
124	1	12	3	13
125	1	12	3	13
126	1	12	3	13
127	1	12	3	13

Table B–4 Failing DIMM Lookup Table (Continued)

	Array 0		Array 2	
Check Bits	DIMM	J#	DIMM	J#
0	0	14	2	15
1	0	14	2	15
2	1	12	3	13
3	1	12	3	13
4	0	14	2	15
5	0	14	2	15
6	1	12	3	13
7	1	12	3	13
8	0	14	2	15
9	0	14	2	15
10	1	12	3	13
11	1	12	3	13
12	0	14	2	15
13	0	14	2	15
14	1	12	3	13
15	1	12	3	13

B.3 EV68 Single-Bit Errors

The procedure for detection down to the set of DIMMs for a single-bit error is very similar to the procedure described in the previous sections. However, you cannot isolate down to a specific data or check bit.

The 21264C (EV68) chip detects and reports a C_ADDR<42:6> failing address that is accurate to the cache block (64 bytes). The syndrome registers (shown in Table B–5) detect data syndrome information, providing isolation down to the low or high quadword of the target octaword that the fault has been detected within. Each of the syndrome registers is able to report 64 data bits (the quadword) and 8 check bits (memory data bus ECC bits).

Table B–5 shows the syndrome hexadecimal to physical data or check bit decoding. For example, if you have an EV68 single-bit C_Syndrome_0 hexadecimal error value equal to 23, the second column indicates the decoded physical data or check bit for this encoding. Use these physical data bits in conjunction with the previously described isolation procedure to isolate the failing DIMMs.

Table B–5 Syndrome to Data Check Bits Table

Syndrome	C_Syndrome 0	C_Syndrome 1
CE	Data Bit 0 or 128	Data Bit 64 or 192
CB	Data Bit 1 or 129	Data Bit 65 or 193
D3	Data Bit 2 or 130	Data Bit 66 or 194
D5	Data Bit 3 or 131	Data Bit 67 or 195
D6	Data Bit 4 or 132	Data Bit 68 or 196
D9	Data Bit 5 or 133	Data Bit 69 or 197
DA	Data Bit 6 or 134	Data Bit 70 or 198
DC	Data Bit 7 or 135	Data Bit 71 or 199
23	Data Bit 8 or 136	Data Bit 72 or 200
25	Data Bit 9 or 137	Data Bit 73 or 201
26	Data Bit 10 or 138	Data Bit 74 or 202
29	Data Bit 11 or 139	Data Bit 75 or 203
2A	Data Bit 12 or 140	Data Bit 76 or 204
2C	Data Bit 13 or 141	Data Bit 77 or 205

Table B–5 Syndrome to Data Check Bits Table (Continued)

Syndrome	C_Syndrome 0	C_Syndrome 1
31	Data Bit 14 or 142	Data Bit 78 or 206
34	Data Bit 15 or 143	Data Bit 79 or 207
0E	Data Bit 16 or 144	Data Bit 80 or 208
0B	Data Bit 17 or 145	Data Bit 81 or 209
13	Data Bit 18 or 146	Data Bit 82 or 210
15	Data Bit 19 or 147	Data Bit 83 or 211
16	Data Bit 20 or 148	Data Bit 84 or 212
19	Data Bit 21 or 149	Data Bit 85 or 213
1A	Data Bit 22 or 150	Data Bit 86 or 214
1C	Data Bit 23 or 151	Data Bit 87 or 215
E3	Data Bit 24 or 152	Data Bit 88 or 216
E5	Data Bit 25 or 153	Data Bit 89 or 217
E6	Data Bit 26 or 154	Data Bit 90 or 218
E9	Data Bit 27 or 155	Data Bit 91 or 219
EA	Data Bit 28 or 156	Data Bit 92 or 220
EC	Data Bit 29 or 157	Data Bit 93 or 221
F1	Data Bit 30 or 158	Data Bit 94 or 222
F4	Data Bit 31 or 159	Data Bit 95 or 223
4F	Data Bit 32 or 160	Data Bit 96 or 224
4A	Data Bit 33 or 161	Data Bit 97 or 225
52	Data Bit 34 or 162	Data Bit 98 or 226
54	Data Bit 35 or 163	Data Bit 99 or 227
57	Data Bit 36 or 164	Data Bit 100 or 228
58	Data Bit 37 or 165	Data Bit 101 or 229
5B	Data Bit 38 or 166	Data Bit 102 or 230
5D	Data Bit 39 or 167	Data Bit 103 or 231
A2	Data Bit 40 or 168	Data Bit 104 or 232
A4	Data Bit 41 or 169	Data Bit 105 or 233
A7	Data Bit 42 or 170	Data Bit 106 or 234
A8	Data Bit 43 or 171	Data Bit 107 or 235
AB	Data Bit 44 or 172	Data Bit 108 or 236
AD	Data Bit 45 or 173	Data Bit 109 or 237

Table B–5 Syndrome to Data Check Bits Table (Continued)

Syndrome	C_Syndrome 0	C_Syndrome 1
B0	Data Bit 46 or 174	Data Bit 110 or 238
B5	Data Bit 47 or 175	Data Bit 111 or 239
8F	Data Bit 48 or 176	Data Bit 112 or 240
8A	Data Bit 49 or 177	Data Bit 113 or 241
92	Data Bit 50 or 178	Data Bit 114 or 242
94	Data Bit 51 or 179	Data Bit 115 or 243
97	Data Bit 52 or 180	Data Bit 116 or 244
98	Data Bit 53 or 181	Data Bit 117 or 245
9B	Data Bit 54 or 182	Data Bit 118 or 246
9D	Data Bit 55 or 183	Data Bit 119 or 247
62	Data Bit 56 or 184	Data Bit 120 or 248
64	Data Bit 57 or 185	Data Bit 121 or 249
67	Data Bit 58 or 186	Data Bit 122 or 250
68	Data Bit 59 or 187	Data Bit 123 or 251
6B	Data Bit 60 or 188	Data Bit 124 or 252
6D	Data Bit 61 or 189	Data Bit 125 or 253
70	Data Bit 62 or 190	Data Bit 126 or 254
75	Data Bit 63 or 191	Data Bit 127 or 255
01	Check Bit 0 or 16	Check Bit 8 or 24
02	Check Bit 1 or 17	Check Bit 9 or 25
04	Check Bit 2 or 18	Check Bit 10 or 26
08	Check Bit 3 or 19	Check Bit 11 or 27
10	Check Bit 4 or 20	Check Bit 12 or 28
20	Check Bit 5 or 21	Check Bit 13 or 29
40	Check Bit 6 or 22	Check Bit 14 or 30
80	Check Bit 7 or 23	Check Bit 15 or 31

Index

E

F

G

H

I

J

T