AlphaServer 1200

User's Guide

Order Number: EK–AS120–UG. A01

This manual is for anyone who manages, operates, or services the
AlphaServer 1200 system. It covers operation, firmware, initial
troubleshooting, and component installation.

Digital Equipment Corporation
Maynard, Massachusetts

Contents

Chapter 4   SRM Console

Chapter 5   AlphaBIOS Console

Chapter 6   Managing the System Remotely

Chapter 7   Installing Components

Chapter 8   Troubleshooting

Index

Examples

Figures

Tables

Preface

Intended Audience

This manual is for anyone who manages, operates, or services an AlphaServer 1200 system.

Document Structure

This manual uses a structured documentation design. Topics are organized into small sections for efficient online and printed reference. Each topic begins with an abstract. You can quickly gain a comprehensive overview by reading only the abstracts. Next is an illustration or example, which also provides quick reference. Last in the structure are descriptive text and syntax definitions.

This manual has eight chapters as follows:

- **Chapter 1, Overview,** describes the system components.

- **Chapter 2, Installing the System,** provides environmental and power requirements, and set-up instructions.

- **Chapter 3, Operation,** gives instructions for powering up the system drawer, booting the operating system, updating firmware.

- **Chapter 4, SRM Console,** describes commands in the SRM console, for DIGITAL UNIX and OpenVMS systems.

- **Chapter 5, AlphaBIOS Console,** describes menu selections in the AlphaBIOS console, for Windows NT systems.

- **Chapter 6, Managing the System Remotely**, provides information on the Remote Console Manager (RCM).

- **Chapter 7, Installing Components,** gives instructions for adding CPU, memory, PCI, and EISA/ISA cards to the system.

- **Chapter 8, Troubleshooting**, provides basic troubleshooting information.

Documentation Titles

Table 1 lists the books in the AlphaServer 1200 documentation set.

Table 1   AlphaServer 1200 Documentation

Title	Order Number
User and Installation Documentation Kit	QZ–011AA–GZ
User's Guide	EK–AS120–UG
Basic Installation	EK–AS120–IG
Service Information	
Service Manual	EK–AS120–SV

Chapter 1

Overview

This chapter provides an overview of the AlphaServer 1200 system features and capabilities. The following topics are covered:

- System Architecture
- System Features
- Front Panel Controls and Indicators
- Rear Panel Ports and Slots
- Console Terminal
- System Options

1.1 System Architecture

The AlphaServer 1200 is a high-performance system. Figure 1–1 shows the system architecture.

Figure 1–1   System Architecture

System Bus

CPU

Memory

SCSI

PCI0 Bus

PCI1 Bus

64-Bit PCI

64-Bit PCI

32-Bit PCI

EISA/ISA Bridge

64-Bit PCI

64-Bit PCI

64-Bit PCI

EISA/ISA Bus

PK-0699-97
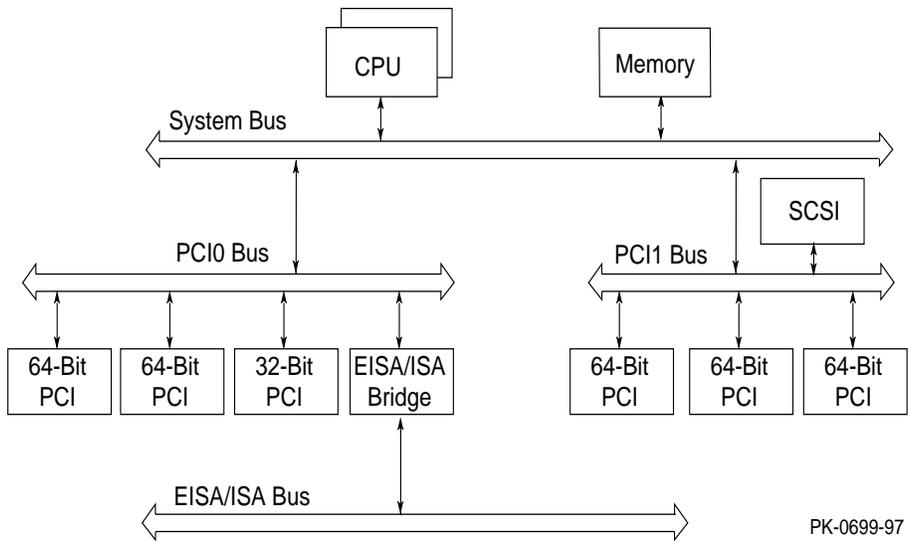
The AlphaServer 1200 pedestal system is housed in an enclosure containing the motherboard, CPU cards, memory modules, and power supplies. This enclosure allows for up to ten internal storage devices: one dedicated diskette drive slot, two removable media slots, and seven 3.5-inch hard disk drives.

The control panel includes Halt, Reset, and On/Off buttons.

All memory and I/O components are on a single board that contains the memory subsystem, the PCI bus, the EISA/ISA bus, the integrated I/O controllers, and the remote console manager (RCM).

Supported Operating Systems

This system supports the following operating systems:

- Microsoft Windows NT
- OpenVMS
- DIGITAL UNIX

While the operating system is running, it controls the system, which is in program mode. When the operating system is *not* running, a second mode, console mode, allows *you* to control system management functions, described in the next section.

System Console Firmware

You perform many of the tasks for managing and configuring your server system in console mode, where the system is controlled by the console subsystem, rather than the operating system.

The console subsystem, located in ROM (read-only memory) on the system board, contains the firmware that interacts directly with hardware components and facilitates interaction between the hardware and the operating system.

1.2 System Features

**The system provides a number of   features that enhance its reliability and
improve its expansion capabilities, as well as facilitate hardware management
and improve security.**

Reliability

- 64-bit Alpha architecture — Provides significantly better performance than 32-bit architecture.

- Error correction code (ECC) on memory and CPU cache — Allows recovery from most cache and memory errors.

- Variable fan speed — Adjusts fan speed according to system temperature.

- Integral Remote Console Management (RCM) function — Enables remote access to system.

- Internal sensors — Monitor and detect internal system temperature, fan failure, power supply temperature.

System Expansion

- Flexible memory architecture — Provides a 128-bit data path with ECC protection. System memory can be upgraded from 64 MB to 2 GB using memory options.

- Six PCI expansion slots and one PCI/EISA expansion slot — Accommodates industry-standard option cards such as Ethernet, FDDI, SCSI, and modems.

- Integrated UltraSCSI controller — Supports tape and CD-ROM without use of an expansion slot.

- Capacity for 10 internal storage devices — Accommodates one StorageWorks modular storage system, which supports up to seven 3½-inch UltraSCSI drives. Also supports up to two 5½-inch, half-height drives (CD-ROM or tape), and one 3½-inch diskette.

- External ports — Two serial ports and one parallel port support external options such as a printer, modem, or local terminal.

• UltraSCSI backplane	Supports high-performance drive technology.
• Hot swap disk capability	Allows replacement of StorageWorks disk drives while the system continues to operate.

System Management

• System diagnostics	Allow local and remote diagnosis of system problems.
• Hardware configuration	Allows local and remote system configuration.
• Unique asset management	Unique system identifier in nonvolatile memory provides easy asset management.
• RAM-based error log	Records startup error messages.
• Firmware upgrade utility	Provides loading and verification of firmware versions.
• Environmental failure events logged in NVRAM	Provides troubleshooting information for system shutdowns.
• Hard drive indicator lights	Provide immediate status information on hard drive activity or failure.

System Security

• Key lock	Limits access to system components.
• Security loop (on rear of system unit)	Allows system to be secured in place.
• Interlock sensor switch	Automatically turns off system power if the top cover is removed while power is on.

1.3 Front Panel Controls and Indicators

The controls and indicators on the front panel of the system unit are shown in Figure 1-1 and described in Table 1-2. The control panel display shows start-up messages during power-up.

Figure 1–2   Front Panel Controls and Indicators
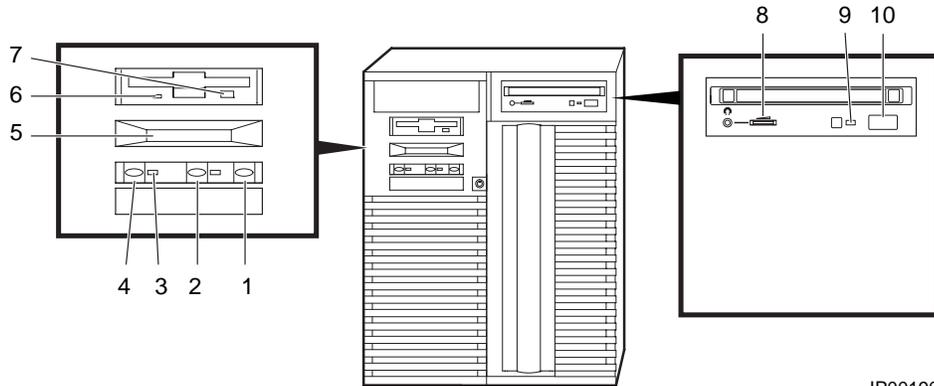
IP00190

Table 1-1 Front Panel Controls and Indicators

Control or Indicator	Function
(1) Halt button	Halts an OpenVMS or DIGITAL UNIX system, returning it to console mode control.
	Does not affect a Windows NT system.
(2) Reset button	Reinitializes the system and performs startup tests. Can be used with the Halt button to bring up the SRM console while in AlphaBIOS.
(3) On/Off indicator	Lights when the system is turned on.
(4) On/Off button	Turns system on and off.
(5) Control panel display	Displays startup messages.
(6) Diskette drive activity indicator	Lights when the system is accessing the diskette drive.
(7) Diskette drive eject button	Ejects a diskette from the diskette drive.
(8) CD-ROM volume control	Adjusts headphone volume.
(9) CD-ROM activity indicator	Lights when the system is accessing the CD-ROM drive.
(10) CD-ROM eject button	Ejects disk from CD-ROM drive.

1.4 Rear Panel Ports and Slots

The ports and slots on the rear of the system unit are shown in Figure 1–3 and described in Table 1-2.

Figure 1–3   Rear Panel Ports and Slots

IP00210
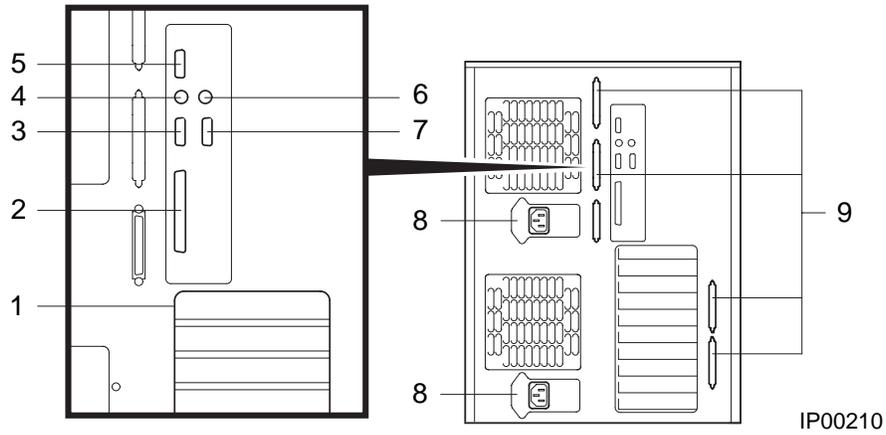
Table 1-2 Rear Panel Ports and Slots

Port or Slot	Used to connect....
(1) Up to six PCI slots and one PCI/EISA slot	Option cards for network, video, or disk controllers
(2) Parallel port	Parallel device such as a printer
(3) Serial port/terminal port (COM2)	Console terminal or serial-line peripheral such as a modem
(4) Mouse port	PS/2-compatible mouse
(5) RCM modem port	External modem for RCM dial-out/access
(6) Keyboard port	PS/2-compatible keyboard
(7) Serial port (COM1)	Serial-line peripheral such as a modem
(8) Power inlets	System unit to power outlets
(9) Four additional SCSI connector knockouts (16-bit, wide)	External SCSI devices without using an adjacent slot

1.5 Console Terminal

The console terminal can be either a serial terminal or a graphics monitor. The power-up display prints to this terminal.

Table 1-3   Comparison of Console Terminals

Condition	Serial Terminal	Graphics Monitor
Does the SRM console power-up display print?	Yes	Yes
Is it supported by the SRM console?	Yes	Yes
Is it supported by the AlphaBIOS console?	Yes	Yes
Is it supported by the DIGITAL UNIX operating system?	Yes	Yes
Is it supported by the OpenVMS operating system?	Yes	Yes
Is it required by the Windows NT operating system?	Yes	Yes

The console terminal can be a serial (character cell) terminal connected to the COM1 port or a graphics monitor connected to a VGA adapter on PCI 0. If the console terminal is connected to COM1, the entire power-up display prints. (See Section 3.2 for information about the power-up display.) If the console terminal is a VGA monitor, console power-up messages are sent to those devices but SROM and XSROM power-up messages are lost.

The **console** environment variable is set to **serial** when the console terminal is a serial terminal; it is set to **graphics** when the console terminal is a graphics monitor. (See Section 4.18 for information about environment variables.)

If the **console** environment variable is set to **serial**, **os_type** is set to **unix** or **openvms**, and no terminal is attached to COM1, pressing a carriage return on a graphics terminal attached to the keyboard port (after power-up testing has completed) makes it the console device and the console prompt is sent to it.

If the **console** environment variable is set to **graphics** and no graphics monitor is attached to the keyboard port, pressing a carriage return on a serial terminal attached to COM1 (after power-up testing has completed) makes it the console device and the console prompt is sent to it.

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

1.6 Options

Options include storage, PCI and EISA/ISA I/O cards, and memory cards. Figure 1–4 shows storage option compartments.

Figure 1–4   Storage Option Compartments
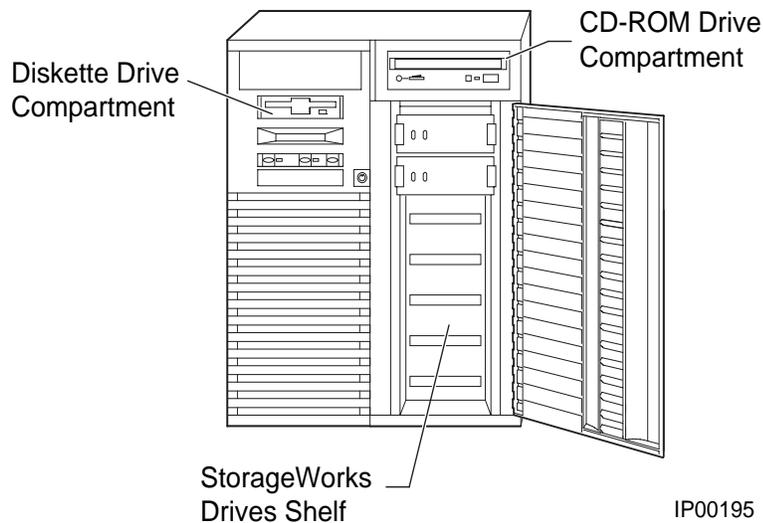
Storage Options

Storage options are located in several compartments inside the system as shown in Figure 1-4. The system accommodates the following types of storage options:

- One diskette drive
- One CD-ROM drive
- Up to seven 3½-inch StorageWorks drives or two 5½-inch drives

PCI and EISA/ISA Options

The system supports PCI options and EISA/ISA options for:

- SCSI storage expansion
- Networking and communications

- Graphics
- Sound

Memory Options

You can increase your system's memory to 2 gigabytes by using various combinations of memory options. Adding more memory allows your system to run memory-intensive software more quickly.

The system supports the following memory option sizes:

64 MB

256 MB

Memory options include a 64-MB option consisting of two 32 MB DIMMs, and a 256-MB option consisting of two 128 MB DIMMs.

Ordering Options

The list of supported options is subject to change. Contact your sales representative for information on the current list of supported options and for information on ordering. If you are an Internet participant, you can obtain information related to the AlphaServer 1200 system through the DIGITAL FTP archive:

ftp.digital.com: /pub/DEC/Alpha/systems/as1200/docs/

For access through the DIGITAL world wide web server:

http://www.service.digital.com/alpha/server/1200.html

Users of the Windows NT operating system can access the Microsoft hardware compatibility list (HCL) of officially supported devices:

http://www.windowsnt.digital.com/support/hcl/hcl.htm

Chapter 2
Installing the System

This chapter explains how to set up and install your system hardware. The following topics are discussed:

- System Setup Overview
- Selecting a Location
- Environmental Requirements
- Power Requirements
- Acoustical Data
- System Accessories
- Connecting the System
- Connecting to Network Hardware
- Locking the System

2.1 System Setup Overview

The following list summarizes the steps for setting up your system. The steps may vary depending on the options in your system.

1. Select a location for the system, giving consideration to service access, environmental requirements, and power requirements.
2. Confirm that you have all the desired accessories that ship with the system and any accessories you may want to add.
3. Connect the keyboard, mouse, printer, and monitor or terminal.
4. Connect to the network hardware.
5. Verify your hardware setup.

2.2 Selecting a Location

When choosing a system location, keep in mind the environmental requirements and power requirements for the system. Figure 2–1 shows the system dimensions and the clearance needed to access the system for servicing.

Figure 2–1 System Dimensions and Service Area

35 cm
(14.1 in)

53 cm
(21 in)

44 cm
(17.4 in)

1 m
(36 in)

65 cm
(26 in)
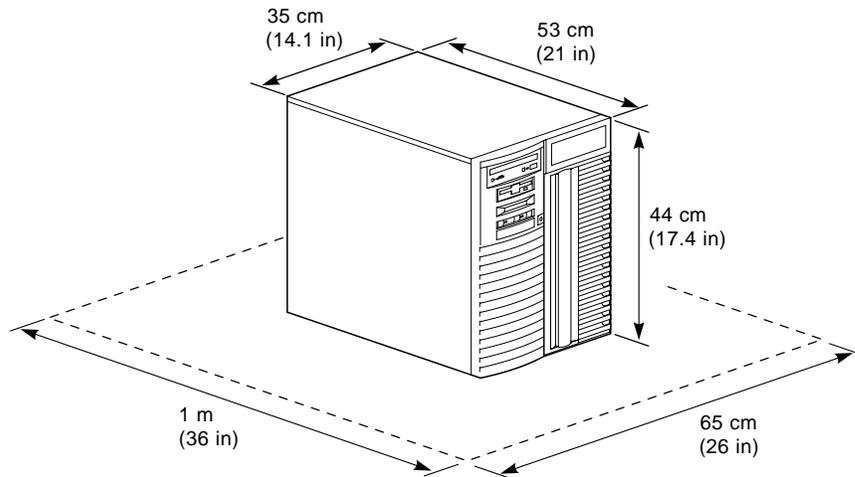
MA00908

2.3 Environmental Requirements

Table 2-1 Environmental Requirements

Condition	Specification
Temperature range	Room temperature: Between 10º C and 40º C (50º F and 104º F).
Relative humidity	Between 10% and 90% (20% to 80% with removable media options).
Air circulation	Allow a minimum clearance of 8 cm (3 inches) on all sides of the system unit to allow sufficient air circulation. Fans inside the system unit circulate the air to prevent excessive heat, which can damage the system components.

2.4 Power Requirements

Your system ships with two power supplies. Both power supplies connect to an AC outlet.

Figure 2–2 Power Supply Requirements

100-120 VAC 7.5A 50-60 Hz
220-240 VAC 3.5A 50-60 Hz

100-120 VAC 7.5A 50-60 Hz
220-240 VAC 3.5A 50-60 Hz

IP00211

= **Properly Grounded Power Receptacle**
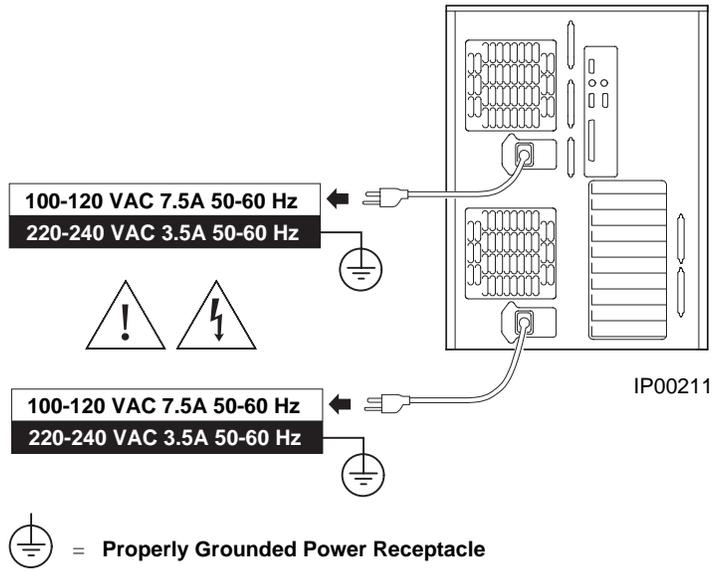
NOTE: Current ratings are maximum with a fully loaded system.

2.5 Acoustical Data

Table 2-2   Acoustics - Declared Values per ISO 9296 and ISO 7779

	Sound Power Level L_{WAd} B		Sound Pressure Level L_{pAm} dBA (bystander positions)	
Product	Idle	Operate	Idle	Operate
54HHA-AA/B, 54KHA-AA/B, and 54ZHA-AA, and all AlphaServer 1200 systems without hard drives	5.6	5.6	37	37
All AlphaServer 1200 systems with 1xRZ1CB disk drive	5.6	5.7	37	38

[Current values for specific configurations are available from Digital Equipment Corporation representatives. 1 B = 10 dBA.]

Table 2-3   Schallemissionswerte - Werteangaben nach ISO 9296 und ISO 7779/DIN EN27779

	Schalleistungspegel L_{WAd} B		Schalldruckpegel L_{pAm} dBA (Zuschauerpositionen)	
Produkt	Leerlauf	Betrieb	Leerlauf	Betrieb
54HHA-AA/B, 54KHA-AA/B, 54ZHA-AA und alle AlphaServer 1200 Systeme ohne Plattenlaufwerke	5,6	5,6	37	37
Alle AlphaServer 1200 Systeme mit 1xRZ1CB Plattenlaufwerk	5,6	5,7	37	38

[Aktuelle Werte für spezielle Ausrüstungsstufen sind über die Digital Equipment Vertretungen erhältlich.   1 B = 10 dBA.]

2.6 System Accessories

Figure 2-3 shows the accessories that are included with the system. Table 2-2 describes the accessories.

Figure 2–3 System Accessories

1 System keys (2)
2 Installation card
3 User's Guide
4 H8571-5 serial connector
5 Mouse
6 Power cord (2)
7 Keyboard
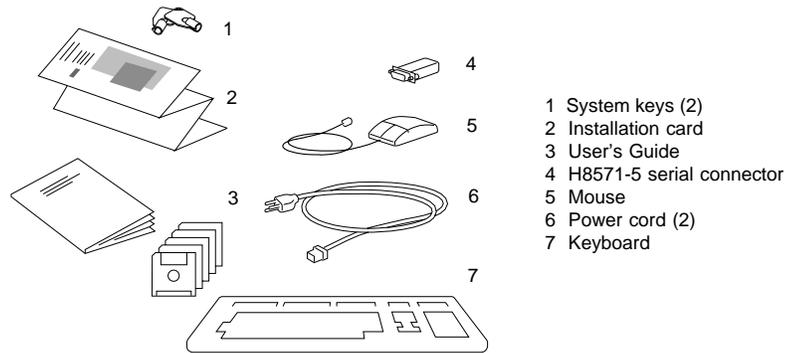
IP00204

If you are using a keyboard, a mouse, or a printer, connect each to the appropriate connector at the rear of the system. See Figure 1-2 to verify the location of the connectors.

2.7 Connecting the System

Connect the system components as shown in Figure 2–4.

Figure 2–4   System Connections
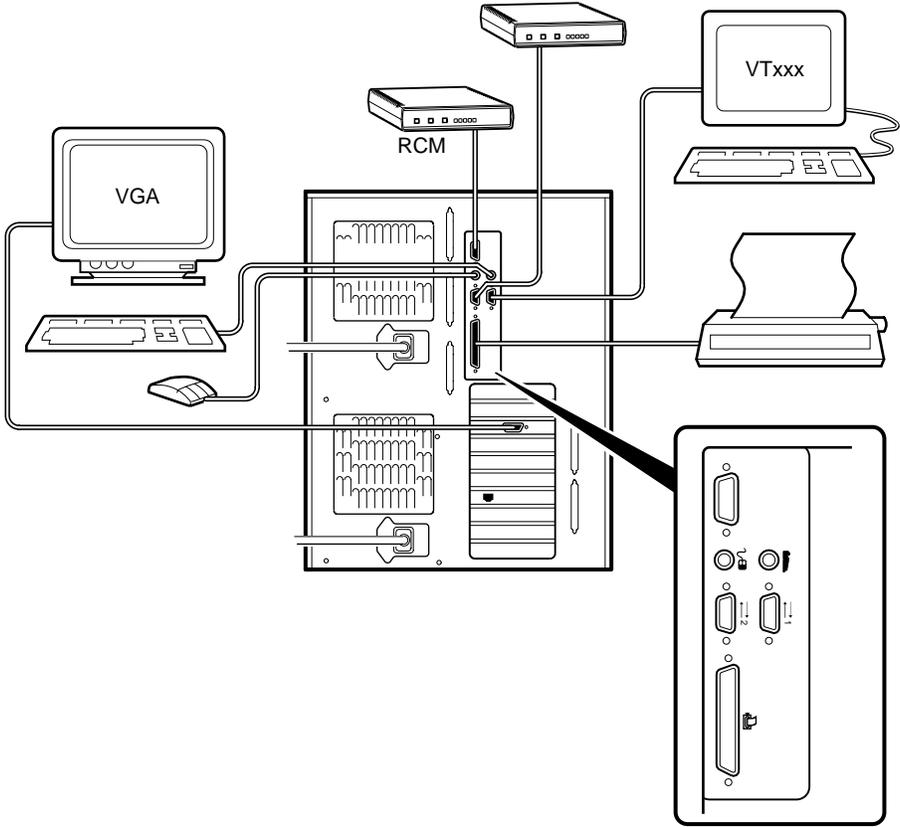
ML014165

2.8 Connecting to Network Hardware

The system supports various network options. You can connect to ThinWire, as shown in Figure 2-5. With appropriate options, you can also connect to FDDI and token ring networks.

Figure 2–5 Network Connections
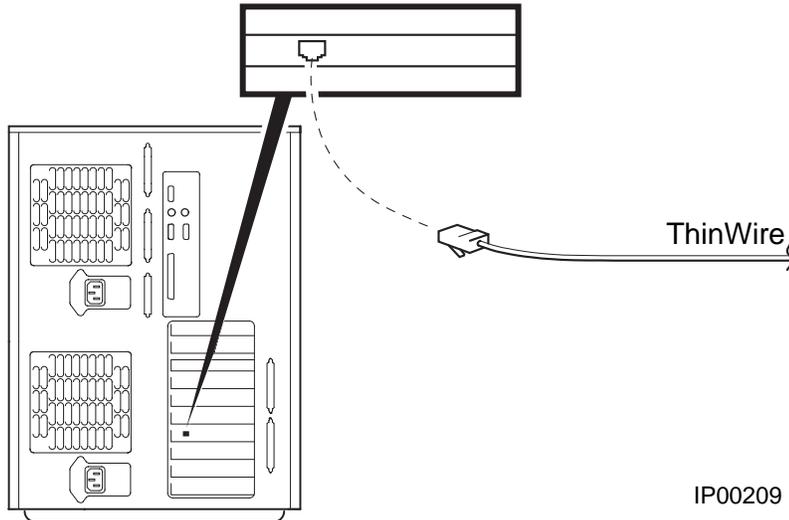
IP00209

2.9 Locking the System

Pedestal systems are protected by a key lock located on the front. Turning the key to the left locks the front door. When the front door is locked, the top cover and side panel latch cannot be accessed.

Figure 2–6 System Lock and Key
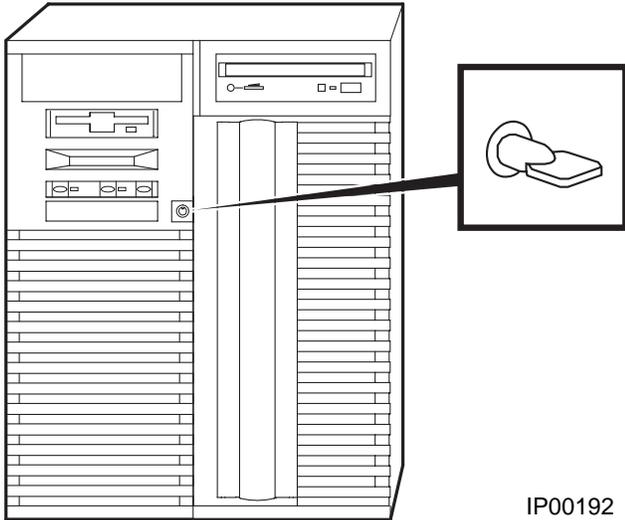
IP00192

Turning the key to the right unlocks the system unit and allows you access to install or remove system components. When the system unit is unlocked, push the lock to open the door. Figure 2–6 shows the system lock in the unlocked position.

Additional security is provided by a latching loop on the rear panel of the system unit that allows you to attach the system unit to a post or other fixed object.

Chapter 3

Operation

This chapter provides basic operating instructions, including powering up the system, booting, and operating system installation. It also provides information about updating firmware.

Sections in this chapter are:

- Powering Up the System
- Power-Up Display
- Booting DIGITAL UNIX
- Installing DIGITAL UNIX
- Booting OpenVMS
- Installing OpenVMS
- Booting Windows NT
- Installing Windows NT
- Updating Firmware
- Hard Disk Partitioning
- Using the Halt Button
- Halt Assertion

3.1   Powering Up the System

To power up the system, press the On/Off button to the On position.

Figure 3–1   Location of On/Off Button

On/Off Button in
("On" Position)

On/Off Button Out
("Off" Position)

IP00193
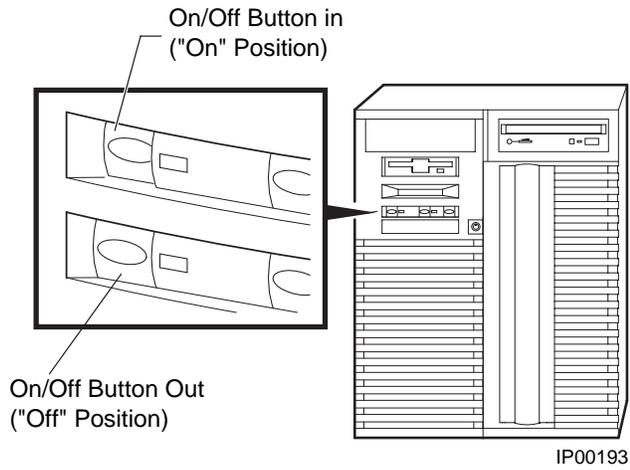
Power up the system by pressing in the On/Off button (see Figure 3–1). Testing begins, and screen text similar to that in Example 3-1 displays (if the console terminal is a serial terminal connected to the COM1 port), along with status messages in the control panel display. If the console terminal is a graphics monitor, only the last few lines of the power-up display print. See Section 3.2 for more information.

3.2   Power-Up Display

The entire power-up display prints to a serial terminal (if the console
environment variable is set to serial); the last several lines print to either a serial
terminal or a graphics monitor.   Power-up status also displays on the control
panel display.

Example 3–1   Power-Up Display

```
P00>>> initialize
Initializing…

 SROM V3.0 on cpu0                                            ❶
 SROM V3.0 on cpu1
XSROM V5.0 on cpu1                                            ❷
XSROM V5.0 on cpu0
BCache testing complete on cpu1                              ❸
BCache testing complete on cpu0
mem_pair0 - 256 MB                                           ❹
20..20..21..21..23..                                         ❺
please wait 9 seconds for T24 to complete
24..24..
Memory testing complete on cpu1
Memory testing complete on cpu0
```

❶ On power-up, the SROM code on each CPU module loads into that module's I-cache and tests the processor chip. If any test fails, power-up terminates.

The primary CPU is determined for the first of three times. The primary CPU then executes a loopback test on each PCI bridge. If this test passes, the bridge LED lights. If it fails, the LED remains clear and power-up continues. The EISA system controller, PCI-to-EISA bridge, COM1 port, and control panel port are all initialized.

Each CPU prints an SROM message to the device attached to the COM1 port and to the control panel display. (The banner prints to the COM1 port if the **console** environment variable is set to **serial**. If it is set to **graphics**, nothing prints to the console terminal, only to the control panel display, until ❻. See Section 4.18 for information about environment variables.)

❷ The S-cache on each CPU module is initialized, and the XSROM code in the system FEPROM is unloaded into them. (If the unload is not successful, the SROM unloads XSROM code from a different FEPROM sector. If the second try fails, the CPU hangs.) Each CPU starts the XSROM code, which prints an XSROM message to the COM1 port and to the control panel display.

❸ The three S-cache banks on each CPU are enabled, and the B-cache is tested. If a failure occurs, a message is printed to the COM1 port and to the control panel display immediately. Each CPU prints a B-cache completion message to COM1.

❹ The primary CPU is determined for the second time. It then sizes memory. Information on memory DIMMs is printed to the COM1 port. If an illegal memory configuration is detected, a warning message is printed to the COM1 port and the control panel display.

❺ Memory is initialized and tested, and test traces are printed to the COM1 port and the control panel display. The numbers for tests 20 and 21 might appear interspersed; this is normal behavior. Test 24 can take several minutes if the memory is very large; the message preceding the test 24 trace indicates an estimate of the time this test will take. If a failure occurs, a message is printed to the COM1 port and to the control panel display immediately. Each CPU prints a test completion message to the COM1 port.

Example 3–1 Power-Up Display (Continued)

```
starting console on CPU 0                                ❻
sizing memory                                            ❼
  0    256 MB DIMM
starting console on CPU 1
probing IOD1 hose 1                                      ❽
  bus 0 slot 1 - NCR 53C810
probing IOD0 hose 0
  bus 0 slot 1 - PCEB
    probing EISA Bridge, bus 1
  bus 0 slot 2 - S3 Trio64/Trio32
  bus 0 slot 3 - DECchip 21040-AA
configuring I/O adapters...
  ncr0, hose 1, bus 0, slot 1
  floppy0, hose 0, bus 1, slot 0
  tulip0, hose 0, bus 0, slot 3
System temperature is 23 degrees C
AlphaServer 1200 Console V5.0-10, 19-NOV-1997 13:57:07 ❾
P00>>>
```

❻ The primary CPU is determined for the last time. The primary CPU unloads PALcode and decompression code from the FEPROM on the system board to its B-cache. The primary CPU then jumps to the PALcode to start the SRM console.

The primary CPU prints a message indicating that it is running the console. Starting with this message, the power-up display is printed to any console terminal, regardless of the state of the **console** environment variable. (If **console** is set to **graphics**, the display from here to the "configuring I/O adapters…" line is saved in a memory buffer an d printed to the graphics monitor after the PCI buses are sized and the VGA device is initialized.)

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

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

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

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

If the system is running the Windows NT operating system (the **os_type** environment variable is set to **nt**), the SRM console loads and starts the AlphaBIOS console.

Refer to Chapter 4 for information about the SRM console and to Chapter 5 for AlphaBIOS.

3.3   Booting DIGITAL UNIX

DIGITAL UNIX® can be booted from a local disk, a disk connected through a cluster, or a remote disk through an Ethernet connection. Refer to the documentation shipped with the operating system for booting instructions.

Example 3–2   Booting DIGITAL UNIX from a Local Disk

```
P00>>> show device                                                   ❶
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1      DKa500                          RRD46   0557
polling ncr1 (NCR 53C810) slot 3, bus 0 PCI, hose 1   SCSI Bus ID 7
dkb100.1.0.3.1      DKb100                          RZ29B   0007
dkb200.2.0.3.1      DKb200                          RZ29B   0007
dkb400.4.0.3.1      DKb400                          RZ29B   0007 ❷
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0     DVA0                            RX23
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1: 08-00-2B-E5-B4-1A


P00>>> boot -file vmunix -flags a dkb400                             ❸
(boot dkb400.4.0.3.1 -file vmunix -flags a)
Building FRU table
block 0 of dkb400.4.0.3.1 is a valid boot block
reading 16 blocks from dkb400.4.0.3.1
bootstrap code read in
base = 200000, image_start = 0, image_bytes = 2000
initializing HWRPB at 2000
initializing page table at 1f2000
initializing machine state
setting affinity to the primary CPU
jumping to bootstrap code

OSF boot - Mon Aug 5 21:56:39 EDT 1997

Loading vmunix ...
.
.
.
```

Example 3–2 Booting DIGITAL UNIX from a Local Disk
(Continued)

```
The system is ready.


DIGITAL UNIX Version 4.0B                                    ❹
```

❶   The **show device** command displays device information, including name and
     type of connection to the system. See Section 4.3 for a description of the **show
     device** command and the device naming convention.

❷   The operating system is on the third disk connected to the system through the
     controller in slot 3 of PCI1. The name of this device, dkb400, is used as an
     argument to the **boot** command.

❸   This command loads DIGITAL UNIX from the disk dkb400, using the boot file
     vmunix and autobooting to multiuser mode. See Section 4.5 for a description
     of the **boot** command.

     The **boot** command accepts the name of a boot device, a boot file name
     through the **-file** option, and boot flags through the **-flags** option. The
     environment variables **bootdef_dev**, **boot_file**, and **boot_osflags** can also be
     used to specify the default boot device or device list, the default boot file, and
     flag information. When an option and the corresponding environment variable
     are both in a command string, the option overrides the environment variable.
     The value of the environment variable, however, is not changed. See Section
     4.18 for information about environment variables.

❹   The operating system banner displays.

Example 3–3   Booting DIGITAL UNIX from a Disk on a Cluster

```
P00>>> show boot*                                               ❶
boot_arg0               0
boot_arg1
boot_dev                dkb0.0.0.5.1
boot_file               vmunix
boot_osflags            a
boot_reset              OFF
bootdef_dev             dkb0.0.0.5.1
booted_dev
booted_file
booted_osflags


P00>>> boot                                                     ❷
(boot dkb0.0.0.5.1 -flags a)
halted CPU 1
halt code = 1
operator initiated halt
PC = fffffc0000264600
Building FRU table
block 0 of dkb0.0.0.5.1 is a valid boot block
reading 16 blocks from dkb0.0.0.5.1
bootstrap code read in
base = 200000, image_start = 0, image_bytes = 2000
initializing HWRPB at 2000
initializing page table at 1f2000
initializing machine state
setting affinity to the primary CPU
Configuring I/O adapters...
jumping to bootstrap code

OSF boot - Mon Jul 29 21:56:39 EDT 1997

Loading vmunix ...

.
.
.

The system is ready.



DIGITAL UNIX Version 4.0B                                       ❸
```

❶ The **show boot*** command lists all environment variables that begin with the string "boot." In this example, the **bootdef_dev** environment variable sets the default boot device to dkb0.0.0.5.1, and **boot_osflags** indicates that the system will autoboot to multiuser mode. See Section 4.18 for descriptions of these environment variables.

❷ The **boot** command needs no options, since the environment variables pass all necessary information.

The **boot** command accepts the name of a boot device, a boot file name through the **-file** option, and boot flags through the **-flags** option. The environment variables **bootdef_dev**, **boot_file**, and **boot_osflags** can also be used to specify the default boot device or device list, the default boot file, and flag information. When an option and the corresponding environment variable are both in a command string, the option overrides the environment variable. The value of the environment variable, however, is not changed. See Section 4.18 for information about environment variables.

❸ The operating system banner displays.

Example 3–4   Booting DIGITAL UNIX from a Remote Disk

```
P00>>> show network                                                   ❶
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1: 08-00-2B-E5-B4-1A                                       ❷
polling tulip1 (DECchip 21041-AA) slot 3, bus 0 PCI, hose 1
ewb0.0.0.3.1: 00-00-F8-00-AC-B4


P00>>> boot -flags an -protocols bootp ewa0                           ❸
(boot ewa0.0.0.4.1 -flags an)
Building FRU table

Trying BOOTP boot.

Broadcasting BOOTP Request...
Received BOOTP Packet File Name: /var/adm/ris/ris0.alpha/hvmunix
local inet address: 16.122.128.26
remote inet address: 16.122.128.59
TFTP Read File Name: /var/adm/ris/ris0.alpha/hvmunix
.....................................................................
.....................................................................
.................
bootstrap code read in
base = 200000, image_start = 0, image_bytes = 9a0fa0
initializing HWRPB at 2000
initializing page table at 1f2000
initializing machine state
setting affinity to the primary CPU
jumping to bootstrap code

Secondary boot program - Thu Aug 1 22:33:13 EST 1996

Loading vmunix ...

.
.
.
The system is ready.

DIGITAL UNIX Version 4.0B                                             ❹
```

❶ The **show network** command displays device information, including name and type of connection to the system. See Section 4.3 for a description of the **show network** command and the device naming convention.

❷ The operating system is on a remote disk accessed through the Ethernet controller in slot 4 of PCI1. The name of this device, ewa0, is used as an argument to the **boot** command.

❸ This command loads DIGITAL UNIX from ewa0, autobooting to multiuser mode. See Section 4.5 for a description of the **boot** command.

The **boot** command accepts the name of a boot device, a boot file name through the **-file** option, and boot flags through the **-flags** option. The environment variables **bootdef_dev**, **boot_file**, and **boot_osflags** can also be used to specify the default boot device or device list, the default boot file, and flag information. When an option and the corresponding environment variable are both in a command string, the option overrides the environment variable. The value of the environment variable, however, is not changed. See Section 4.18 for information about environment variables.

❹ The operating system banner displays.

3.4   Installing DIGITAL UNIX

**DIGITAL UNIX is installed from the CD-ROM. Refer to the documentation
shipped with the CD-ROM for installation instructions.**

Example 3–5   Installing DIGITAL UNIX

```
P00>>> boot dka500                                            ❶
(boot dka500.5.0.1.1 –flags a)
Building FRU table                                            ❷
block 0 of dka500.5.0.1.1 is a valid boot block
reading 16 blocks from dka500.5.0.1.1
bootstrap code read in
base = 200000, image_start = 0, image_bytes = 2000
initializing HWRPB at 2000
initializing page table at 1f2000
initializing machine state
setting affinity to the primary CPU
jumping to bootstrap code

OSF boot - Mon Jul 24 21:56:39 EDT 1995

Loading hupdate ...
Starting hupdate at 0x20000000
Using hvmunix

Loading hvmunix ...
Current PAL Revision <0x20000000010111>
Switching to OSF PALcode Succeeded
New PAL Revision <0x20000000020115>
Loading into KSEG Address Space

Sizes:
text = 4460352
data = 5555024
bss  = 1498368
Starting at 0xfffffc0000239290
```

Continued on next page

Example 3–5   Installing DIGITAL UNIX (Continued)

```
Alpha boot: available memory from 0x2dfe000 to 0x7fff6000
physical memory = 2048.00 megabytes.
available memory = 2001.96 megabytes.
using 7856 buffers containing 61.37 megabytes of memory

Firmware revision: 5.0
PALcode: DIGITAL-UNIX/OSF version 1.21-19
AlphaServer 1200
pci1 at mcbus0 slot 5
psiop0 at pci1 slot 1
.
.
.
Printer service started
The system is ready.



DIGITAL UNIX Version 4.0B

login: root
Password:
```

❶   Use the **boot** command to install the operating system from the CD-ROM, which is always dka500.

❷   See your operating system documentation for installation instructions.

3.5   Booting OpenVMS

OpenVMS can be booted from a local disk, a disk connected through a cluster, or a remote disk through an Ethernet connection. Refer to the documentation shipped with the operating system for booting instructions.

Example 3–6   Booting OpenVMS from a Local Disk

```
P00>>> show device                                              ❶
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1      DKa500                          RRD46   0557
polling ncr1 (NCR 53C810) slot 2, bus 0 PCI, hose 1   SCSI Bus ID 7
dkb0.0.0.2.1        DKb0                            RZ29B   0016
dkb100.1.0.2.1      DKb100                          RZ29B   0016
.
.
.
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0     DVA0                            RX23
polling tulip0 (DECchip 21040-AA) slot 3, bus 0 PCI, hose 0
ewa0.0.0.3.0: 00-00-F8-20-14-61
P00>>> show boot_reset                                          ❷
boot_reset              ON
P00>>> show bootdef_dev                                         ❸
bootdef_dev             dkb0.0.0.2.1
P00>>> boot                                                     ❹
 SROM V3.0 on cpu0                                              ❷
 SROM V3.0 on cpu1
XSROM V5.0 on cpu1
XSROM V5.0 on cpu0
.
.
.
System temperature is 24 degrees C
AlphaServer 1200 Console V5.0-1, 20-SEP-1997 15:17:48
```

Continued on next page

3-16   User's Guide

Example 3–6   Booting OpenVMS from a Local Disk (Continued)

```
CPU 0 booting

(boot dkb0.0.0.2.1 -flags 0)
Building FRU table
block 0 of dkb0.0.0.2.1 is a valid boot block
reading 1002 blocks from dkb0.0.0.2.1
bootstrap code read in
.
.
.
Welcome to OpenVMS Alpha (TM) Operating System, Version V7.1x ❺
```

❶   The **show device** command displays device information, including name and type of connection to the system. See Section 4.3 for a description of the **show device** command and the device naming convention.

❷   The **boot_reset** environment variable was previously set to "on," causing the power-up trace to display when the system initializes (see Section 3.2). See Section 4.18 for commands used with environment variables.

❸   The **bootdef_dev** environment variable specifies the default boot device. In this example, the default boot device was previously set to dkb0.0.0.2.1.

❹   No boot device is specified in the **boot** command; the default boot device was set with the environment variable. See Section 4.5 for a description of the **boot** command.

      The **boot** command accepts the name of a boot device, a boot file name through the **-file** option, and boot flags through the **-flags** option. The environment variables **bootdef_dev**, **boot_file**, and **boot_osflags** can also be used to specify the default boot device or device list, the default boot file, and flag information. When an option and the corresponding environment variable are both in a command string, the option overrides the environment variable. The value of the environment variable, however, is not changed. See Section 4.18 for information about environment variables.

❺   The operating system banner displays.

Example 3–7   Booting OpenVMS from a Disk on a Cluster

```
P00>>> show bootdef_dev                                      ❶
bootdef_dev               dua1.8.0.4.1
P00>>> show device                                           ❷
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1    SCSI Bus ID 7
dka500.5.0.1.1      DKa500                      RRD45  1645
polling dac0 (Mylex DAC960) slot 3, bus 0 PCI, hose 1
dra0.0.0.3.1        DRA0                         1 Member JBOD
dra1.0.0.3.1        DRA1                         1 Member JBOD
dra2.0.0.3.1        DRA2                         4 Member RAID 5
polling cipca0 (CIPCA) slot 4, bus 0 PCI, hose 1    cixcd_poll_comp
cipca_a.4.0.4.1     dua      CI Bus ID 4                      ❸
dua24.2.0.4.1       $22$DUA24 (HSC702)           RA92
dua30.2.0.4.1       $22$DUA30 (HSC702)           RA92
.
.
.
dua0.8.0.4.1        $22$DUA0 (RAWHSJ)            HSX1
dua1.8.0.4.1        $22$DUA1 (RAWHSJ)            HSX1          ❸
.
.
.
P00>>> boot                                                  ❹
(boot dua1.8.0.4.1 -flags 0)
Building FRU table
.
.
.
Welcome to OpenVMS Alpha (TM) Operating System, Version V7.1x ❺
```

❶ The **bootdef_dev** environment variable specifies the default boot device. In this example, the default boot device was previously set to dua1.8.0.4.1. See Section 4.18 for commands used with environment variables.

❷ The **show device** command displays device information, including name and type of connection to the system. See Section 4.3 for a description of the **show device** command and the device naming convention.

❸ The disk dua1.8.0.4.1 is on the CI cluster that includes this system.

❹ No boot device is specified in the **boot** command; the default boot device was set with the environment variable. See Section 4.5 for a description of the **boot** command.

The **boot** command accepts the name of a boot device, a boot file name through the **-file** option, and boot flags through the **-flags** option. The environment variables **bootdef_dev**, **boot_file**, and **boot_osflags** can also be used to specify the default boot device or device list, the default boot file, and flag information. When an option and the corresponding environment variable are both in a command string, the option overrides the environment variable. The value of the environment variable, however, is not changed. See Section 4.18 for information about environment variables.

❺ The operating system banner prints.

Example 3–8 Booting OpenVMS from a Remote Disk

```
P00>>> show device                                                    ❶
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1      DKa500                        RRD45   1645
polling kzpsa0 (DEC KZPSA) slot 2, bus 0 PCI, hose 1   TPwr 1 Fast 1 Bus ID 5
kzpsa0.5.0.2.1      dkb        TPwr 1 Fast 1 Bus ID 5    N01   A10
polling kzpsa1 (DEC KZPSA) slot 3, bus 0 PCI, hose 1   TPwr 1 Fast 1 Bus ID 5
kzpsa1.5.0.3.1      dkc        TPwr 1 Fast 1 Bus ID 5    N01   A10
polling kzpsa2 (DEC KZPSA) slot 4, bus 0 PCI, hose 1   TPwr 1 Fast 1 Bus ID 5
kzpsa2.5.0.4.1      dkd        TPwr 1 Fast 1 Bus ID 5    P01   A10
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0     DVA0                        RX23
polling tulip0 (DECchip 21140-AA) slot 2, bus 0 PCI, hose 0
ewa0.0.0.2.0: 00-00-F8-00-42-04                                       ❶
P00>>> boot ewa0 -flags 0                                             ❷
(boot ewa0.0.0.2.0 -flags 0)
Building FRU table

Trying MOP boot.
.............

Network load complete.
.
.
.
Welcome to OpenVMS Alpha (TM) Operating System, Version V7.1x ❸
```

❶   The **show device** command displays device information, including name and
     type of connection to the system. In this example the Ethernet connection is
     ewa0. See Section 4.3 for a description of the **show device** command and the
     device naming convention.

❷   The **boot** command specifies ewa0 as the boot device. See Section 4.5 for a
     description of the **boot** command.

     The **boot** command accepts the name of a boot device, a boot file name
     through the **-file** option, and boot flags through the **-flags** option. The
     environment variables **bootdef_dev**, **boot_file**, and **boot_osflags** can also be
     used to specify the default boot device or device list, the default boot file, and
     flag information. When an option and the corresponding environment variable
     are both in a command string, the option overrides the environment variable.
     The value of the environment variable, however, is not changed. See Section
     4.18 for information about environment variables.

❸   The operating system banner prints.

3.6 Installing OpenVMS

OpenVMS is installed from the CD-ROM. Refer to the documentation shipped with the CD-ROM for complete installation instructions.

Example 3–9 Installing OpenVMS

```
P00>>> boot -flags 0,0 dka500                                          ❶
Initializing...
SROM V3.0 on cpu0
.
.  [The initialization display prints. See Section 3.2.]
.

AlphaServer 1200 Console V5.0-2, 2-APR-1997 15:17:48

CPU 0 booting

(boot dka500.5.0.1.1 -flags 0,0)
Building FRU table
block 0 of dka500.5.0.1.1 is a valid boot block
reading 1002 blocks from dka500.5.0.1.1
bootstrap code read in
base = 200000, image_start = 0, image_bytes = 7d400
initializing HWRPB at 2000
initializing page table at 1f2000
initializing machine state
setting affinity to the primary CPU
jumping to bootstrap code

    OpenVMS (TM) Alpha Operating System, Version 7.1x               ❷

%SMP-I-SECMSG, CPU #01 message:    P01>>>START
%SMP-I-CPUBOOTED, CPU #01 has joined the PRIMARY CPU in
multiprocessor operation
```

Continued on next page

Example 3–9 Installing OpenVMS (Continued)

```
Installing required known files...

    Configuring devices...


*************************************************************

 You can install or upgrade the OpenVMS Alpha operating system
 or you can install or upgrade layered products that are included
 on the OpenVMS Alpha operating system CD-ROM.

 You can also execute DCL commands and procedures to perform
 "standalone" tasks, such as backing up the system disk.

 Please choose one of the following:

     1)  Install or upgrade OpenVMS Alpha Version 7.1x
     2)  List layered product kits that this procedure can install
     3)  Install or upgrade layered product(s)
     4)  Execute DCL commands and procedures
     5)  Shut down this system

Enter CHOICE or ? to repeat menu: (1/2/3/4/5/?)
```

❶   Use the **boot** command to install the operating system from the CD-ROM,
     which is always dka500.

❷   See your operating system documentation for installation instructions.

3.7   Booting Windows NT

Microsoft Windows NT is started from the AlphaBIOS Boot screen.

Figure 3–2   AlphaBIOS Boot Screen
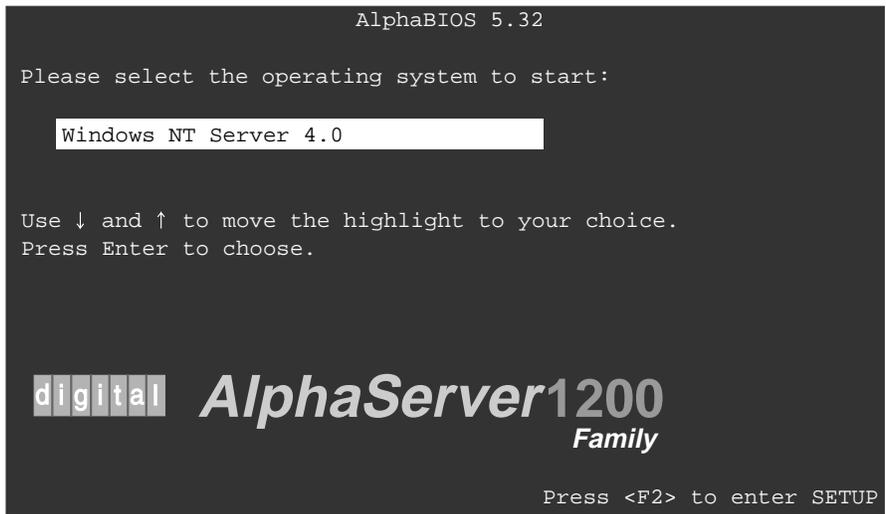
PK-0728C-96

Two SRM environment variables must be set properly for Windows NT to boot. The setting of the SRM **os_type** environment variable determines if AlphaBIOS is loaded and started on reset and power-up. If **os_type** is set to **nt**, after the power-up display the SRM console is loaded and started, and it then loads and starts the AlphaBIOS console. AlphaBIOS must be running before Windows NT can be booted. Windows NT requires a graphics monitor as its console. Setting the SRM **console** environment variable to **graphics** and having a graphics monitor attached to your system meets this requirement. After setting these two variables, you have to power-down and power-up your system for them to take effect.

The method used for booting Windows NT is determined by the setting of **Auto Start** in the AlphaBIOS **Standard CMOS Setup** screen (see Chapter 5).

- If **Auto Start** is enabled, the primary version of Windows NT starts automatically.

- If **Auto Start** is disabled, use the arrow keys to select the Windows NT version to start. Press Enter to boot Windows NT.

*NOTE: The SRM **console** environment variable must be set to **graphics** before booting Windows NT, though this setting is not necessary to run AlphaBIOS.*

3.8   Installing Windows NT

Windows NT is installed from the CD-ROM. Insert the CD-ROM into the drive, start AlphaBIOS Setup, select the menu item Install Windows NT, and follow the prompts.

Figure 3–3   Installing Windows NT

```
                                    AlphaBIOS Setup

   ┌──────────────────────────────────────────┐
   │ Display System Configuration...           │
   │ Upgrade AlphaBIOS                          │
   │ Hard Disk Setup...                         │
   │ CMOS Setup...                              │
   │ Install Windows NT                         │
   │ Utilities                              ►   │
   │ About AlphaBIOS...                         │
   └──────────────────────────────────────────┘


   ┌──────────────────────────────────────────┐
   │ Press ENTER to install Windows NT.        │
   │                                            │
   │                                            │
   └──────────────────────────────────────────┘

   ESC=Exit
```
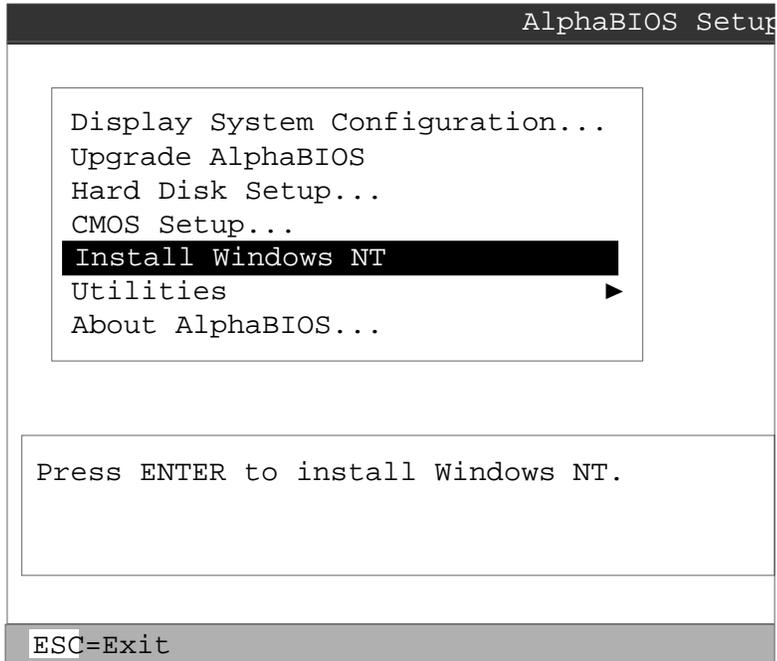
PK-0726B-96

Windows NT requires a partitioned and formatted hard disk drive. If your drive is not partitioned or formatted, follow the instructions in Section 3.10 before installing the Windows NT operating system.

Up to four versions of Windows NT can be resident in a system at one time.

If this is a new Windows NT installation, start with this procedure:

1. Start AlphaBIOS. If the system is in the SRM console, issue the command **alphabios**.

2. From the **AlphaBIOS Boot** screen, enter **AlphaBIOS Setup** by pressing the F2 key.

3. From **AlphaBIOS Setup** select **CMOS Setup**, and press Enter.

4. From **CMOS Setup** set the system date and time.

5. Return to the **AlphaBIOS Setup** screen by pressing the F10 key.

6. Select **Hard Disk Setup**, and press Enter.

7. Perform an express hard disk setup by pressing the F7 key to enter **Express Setup**.

8. Continue the setup by pressing the F10 key.

9. Go to the procedure below.

This procedure is for all Windows NT installations:

1. Put the Windows NT CD into the CD-ROM drive.

2. Start **AlphaBIOS Setup**.

3. Select **Install Windows NT** and press Enter.

4. Follow the prompts to complete the installation. For more information on installing Windows NT, refer to the *Installation Guide* in your Windows NT software package.

3.9   Updating Firmware

Start the Loadable Firmware Update (LFU) utility by issuing the lfu command at the SRM console prompt, booting it from the CD-ROM while in the SRM console, or selecting Update AlphaBIOS in the AlphaBIOS Setup screen.

Example 3–10   Starting LFU from the SRM Console

```
P00>>> lfu

          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: cda0
.
.
.
UPD>
```

Figure 3–4   Starting LFU from the AlphaBIOS Console


```
┌──────────────────────────────────────────────────────────────┐
│                         AlphaBIOS Setup                        │
├──────────────────────────────────────────────────────────────┤
│   ┌──────────────────────────────────────────┐                │
│   │ Display System Configuration...           │                │
│   │ Upgrade AlphaBIOS                          │                │
│   │ Hard Disk Setup                            │                │
│   │ CMOS Setup...                              │                │
│   │ Install Windows NT                         │                │
│   │ Utilities                              ►   │                │
│   │ About AlphaBIOS...                         │                │
│   └──────────────────────────────────────────┘                │
│                                                                │
│   ┌──────────────────────────────────────────────────────────┐│
│   │ Press ENTER to upgrade your AlphaBIOS from floppy or CD-ROM.││
│   │                                                            ││
│   └──────────────────────────────────────────────────────────┘│
│                                                                │
│  ESC=Exit                                                      │
└──────────────────────────────────────────────────────────────┘
```
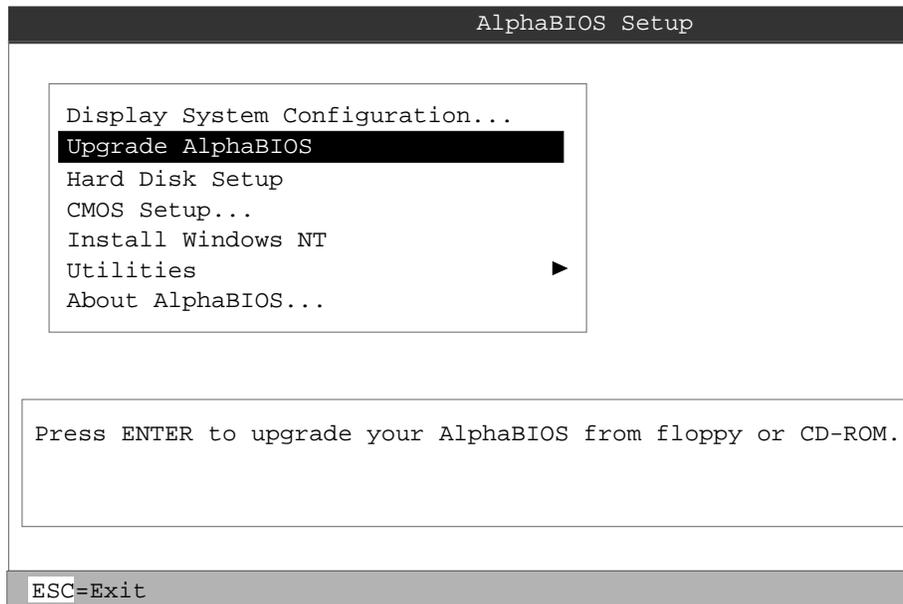
PK-0726A-96

Use the Loadable Firmware Update (LFU) utility to update system firmware.

You can start LFU from either the SRM console or the AlphaBIOS console.

- From the SRM console, start LFU by issuing the **lfu** command (see Example 3–10). Also from the SRM console, LFU can be booted from the Alpha CD-ROM (V3.7 or later), as shown in Example 3–11.

- From the AlphaBIOS console, select **Update AlphaBIOS** from the **AlphaBIOS Setup** screen (see Figure 3–4).

A typical update procedure is:

1. Start LFU.

2. Use the LFU **list** command to show the revisions of modules that LFU can update and the revisions of update firmware.

3. Use the LFU **update** command to write the new firmware.

4. Use the LFU **exit** command to go back to the console.

The sections that follow show examples of updating firmware from the local CD-ROM, the local floppy, and a network device. Following the examples is an LFU command reference.

Example 3–11 Booting LFU from the CD-ROM

```
P00>>> show dev ncr0
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1     DKa500                    RRD46  1645
P00>>> boot dka500
(boot dka500.5.0.1.1 -flags 0,0)
block 0 of dka500.5.0.1.1 is a valid boot block
.
.
.
jumping to bootstrap code

The default bootfile for this platform is
        [AS1200]AS1200_LFU.EXE
Hit <RETURN> at the prompt to use the default bootfile.
Bootfile: <CR>

Starting Firmware Update Utility

        ***** Loadable Firmware Update Utility *****
.
.
.
UPD>
```

3.9.1 Updating Firmware from the CD-ROM

Insert the Alpha CD-ROM, start LFU, and select cda0 as the load device.

Example 3–12 Updating Firmware from the CD-ROM

```
          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: cda0        ❶

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS1200FW]: AS1200CP  ❷

Copying AS1200CP from DKA500.5.0.1.1 .
Copying [as1200]TCREADME from DKA500.5.0.1.1 .
Copying [as1200]TCSRMROM from DKA500.5.0.1.1 ...................
Copying [as1200]TCARCROM from DKA500.5.0.1.1 .............

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

UPD> list                                                        ❹

Device           Current Revision    Filename      Update Revision
AlphaBIOS           V5.32-0          arcrom           V6.40-1
srmflash            V5.0-1           srmrom           V6.0-3
```

Continued on next page

❶ Select the device from which firmware will be loaded. The choices are the internal CD-ROM, the internal floppy disk, or a network device. In this example, the internal CD-ROM is selected.

❷ Select the file that has the firmware update, or press Enter to select the default file. The file options are:

AS1200FW
(default)   SRM console, AlphaBIOS console, and I/O adapter firmware.

AS1200CP   SRM console and AlphaBIOS console firmware only.

AS1200IO   I/O adapter firmware only.

In this example the file for console firmware (AlphaBIOS and SRM) is selected.

❸ The LFU function table and prompt (UPD>) display.

❹ Use the LFU **list** command to determine the revision of firmware in a device and the most recent revision of that firmware available in the selected file. In this example, the resident firmware for each console (SRM and AlphaBIOS) is at an earlier revision than the firmware in the update file.

Continued on next page

Example 3–12 Updating Firmware from the CD-ROM (Continued)

```
UPD> update *                                                    ❺
WARNING: updates may take several minutes to complete for each
device.

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

                        DO NOT ABORT!
AlphaBIOS      Updating to V6.40-1... Verifying V6.40-1... PASSED.

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

                        DO NOT ABORT!
srmflash        Updating to V6.0-3...  Verifying V6.0-3...
PASSED.

UPD> exit                                                        ❼
```

❺   The **update** command updates the device specified or all devices. In this
     example, the wildcard indicates that all devices supported by the selected
     update file will be updated.

❻   For each device, you are asked to confirm that you want to update the
     firmware. The default is no. Once the update begins, do not abort the
     operation. Doing so will corrupt the firmware on the module.

❼   The **exit** command returns you to the console from which you entered LFU
     (either SRM or AlphaBIOS).

3.9.2 Updating Firmware from Floppy Disk — Creating the Diskettes

---

Create the update diskettes before starting LFU. See Section 3.9.3 for an example of the update procedure.

Table 3–1   File Locations for Creating Update Diskettes on a PC

Console Update Diskette	I/O Update Diskette
AS1200FW.TXT	AS1200IO.TXT
AS1200CP.TXT	TCREADME.SYS
TCREADME.SYS	CIPCA315.SYS
TCSRMROM.SYS	DFPAA310.SYS
TCARCROM.SYS	KZPAAA11.SYS

To update system firmware from floppy disk, you first must create the firmware update diskettes. You will need to create two diskettes: one for console updates, and one for I/O.

1. Download the update files from the Internet.
2. On a PC, copy files onto two FAT-formatted diskettes as shown in Table 3–1.
3. From an OpenVMS system, copy files onto two ODS2-formatted diskettes as shown in Example 3–13.

Example 3–13   Creating Update Diskettes on an OpenVMS System

Console update diskette

```
$ inquire ignore "Insert blank HD floppy in DVA0, then continue"
$ set verify
$ set proc/priv=all
$ init /density=hd/index=begin dva0: tcods2cp
$ mount dva0: tcods2cp
$ create /directory dva0:[as1200]
$ copy tcreadme.sys dva0:[as1200]tcreadme.sys
$ copy as1200fw.txt dva0:[as1200]as1200fw.txt
$ copy as1200cp.txt dva0:[as1200]as1200cp.txt
$ copy tcsrmrom.sys dva0:[as1200]tcsrmrom.sys
$ copy tcarcrom.sys dva0:[as1200]tcarcrom.sys
$ dismount dva0:
$ set noverify
$ exit
```

I/O update diskette

```
$ inquire ignore "Insert blank HD floppy in DVA0, then continue"
$ set verify
$ set proc/priv=all
$ init /density=hd/index=begin dva0: tcods2io
$ mount dva0: tcods2io
$ create /directory dva0:[as1200]
$ create /directory dva0:[options]
$ copy tcreadme.sys dva0:[as1200]tcreadme.sys
$ copy as1200fw.txt dva0:[as1200]as1200fw.txt
$ copy as1200io.txt dva0:[as1200]as1200io.txt
$ copy cipca214.sys dva0:[options]cipca214.sys
$ copy dfpaa246.sys dva0:[options]dfpaa246.sys
$ copy kzpsaA10.sys dva0:[options]kzpsaa10.sys
$ dismount dva0:
$ set noverify
$ exit
```

3.9.3   Updating Firmware from Floppy Disk — Performing the Update

Insert an update diskette (see Section 3.9.2) into the floppy drive. Start LFU and select dva0 as the load device.

Example 3–14   Updating Firmware from the Floppy Disk

```
          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: dva0        ❶

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS1200IO,(AS1200CP)]:
AS1200IO                                                         ❷

Copying AS1200IO from DVA0 .
Copying TCREADME from DVA0 .
Copying CIPCA214 from DVA0 .
Copying DFPAA252 from DVA0 ...
Copying KZPSAA11 from DVA0 ...
.
.  [The function table displays, followed by the UPD> prompt, as
.  shown in Example 3-12.]

UPD> list                                                        ❸

Device           Current Revision   Filename        Update Revision
AlphaBIOS          V5.12-3           arcrom          Missing file
pfi0                2.46             dfpaa_fw           2.52
srmflash           T3.2-21           srmrom          Missing file
                                     cipca_fw          A214
                                     kzpsa_fw           A11
```

Continued on next page

❶   Select the device from which firmware will be loaded. The choices are the internal CD-ROM, the internal floppy disk, or a network device. In this example, the internal floppy disk is selected.

❷   Select the file that has the firmware update, or press Enter to select the default file. When the internal floppy disk is the load device, the file options are:

AS1200CP (default)   SRM console and AlphaBIOS console firmware only.

AS1200IO   I/O adapter firmware only.

The default option in Example 3–12 (AS1200FW) is not available, since the file is too large to fit on a 1.44 MB diskette. This means that when a floppy disk is the load device, you can update either console firmware or I/O adapter firmware, but not both in the same LFU session. If you need to update both, after finishing the first update, restart LFU with the **lfu** command and insert the diskette with the other file.

In this example the file for I/O adapter firmware is selected.

❸   Use the LFU **list** command to determine the revision of firmware in a device and the most recent revision of that firmware available in the selected file. In this example, the update revision for console firmware displays as "Missing file" because only the I/O firmware files are available on the floppy disk.

Example 3–14 Updating Firmware from the Floppy Disk (Continued)

```
UPD> update pfi0                                              ❹
WARNING: updates may take several minutes to complete for each device.

Confirm update on: pfi0               [Y/(N)] y               ❺

                      DO NOT ABORT!
pfi0              Updating to 3.10... Verifying to 3.10... PASSED.

UPD> lfu                                                      ❻

         ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: dva0

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS1200IO,(AS1200CP)]:❼

.
. (The function table displays, followed by the UPD> prompt.
.  Console firmware can now be updated.)

UPD> exit                                                    ❽
```

❹   The **update** command updates the device specified or all devices.

❺   For each device, you are asked to confirm that you want to update the firmware. The default is no. Once the update begins, do not abort the operation. Doing so will corrupt the firmware on the module.

❻   The **lfu** command restarts the utility so that console firmware can be updated. (Another method is shown in Example 3–15, where the user specifies the file AS1200FW and is prompted to insert the second diskette.)

❼   The default update file, AS1200CP, is selected. The console firmware can now be updated, using the same procedure as for the I/O firmware.

❽   The **exit** command returns you to the console from which you entered LFU (either SRM or AlphaBIOS).

Example 3–15   Selecting AS1200FW to Update Firmware from the Internal Floppy Disk

```
P00>>> lfu

        ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: dva0

Please enter the name of the firmware files list, or
Press <return> to use the default filename [AS1200IO,(AS1200CP)]: as1200fw

Copying AS1200FW from DVA0 .
Copying TCREADME from DVA0 .
Copying TCSRMROM from DVA0 .........................
Copying TCARCROM from DVA0 ..............
Copying CIPCA214 from DVA0
Please insert next floppy containing the firmware,
Press <return> when ready. Or type DONE to abort.
Copying CIPCA214 from DVA0 .
Copying DFPAA246 from DVA0 ...
Copying KZPSAA10 from DVA0 ...
.
.
.
```

3.9.4   Updating Firmware from a Network Device

Copy files to the local MOP server's MOP load area, start LFU, and select ewa0 as the load device.

Example 3–16   Updating Firmware from a Network Device

```
          ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: ewa0        ❶

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS1200FW]:           ❷

Copying AS1200FW from EWA0 .
Copying TCREADME from EWA0 .
Copying TCSRMROM from EWA0 ..........................
Copying TCARCROM from EWA0 ...........
Copying CIPCA214 from EWA0 .
Copying DFPAA246 from EWA0 ...
Copying KZPSAA11 from EWA0 ...


.
.
.   [The function table displays, followed by the UPD> prompt, as
.   shown in Example 3–12.]

UPD> list                                                        ❸

Device           Current Revision    Filename        Update Revision
AlphaBIOS          V5.32-0           arcrom             V6.40-1
kzpsa0             A10               kzpsa_fw           A11
kzpsa1             A10               kzpsa_fw           A11
srmflash           V5.0-1            srmrom             V6.0-3
                                     cipca_fw           A214
                                     dfpaa_fw             2.46
```

Continued on next page

Before starting LFU, download the update files from the Internet. You will need the files with the extension .SYS. Copy these files to your local MOP server's MOP load area.

❶   Select the device from which firmware will be loaded. The choices are the CD-ROM, the internal floppy disk, or a network device. In this example, a network device is selected.

❷   Select the file that has the firmware update, or press Enter to select the default file. The file options are:

AS1200FW (default)   SRM console, AlphaBIOS console, and I/O adapter firmware.

AS1200CP   SRM console and AlphaBIOS console firmware only.

AS1200IO   I/O adapter firmware only.

In this example the default file, which has both console firmware (AlphaBIOS and SRM) and I/O adapter firmware, is selected.

❸   Use the LFU **list** command to determine the revision of firmware in a device and the most recent revision of that firmware available in the selected file. In this example, the resident firmware for each console (SRM and AlphaBIOS) and I/O adapter is at an earlier revision than the firmware in the update file.

Example 3–16   Updating Firmware from a Network Device (Continued)

```
UPD> update * -all                                               ❹
WARNING: updates may take several minutes to complete for each
device.

                          DO NOT ABORT!
AlphaBIOS     Updating to V6.40-1... Verifying V6.40-1...  PASSED.

                          DO NOT ABORT!
kzpsa0        Updating to  A11    ...  Verifying A11...  PASSED.

                          DO NOT ABORT!
kzpsa1        Updating to  A11    ...  Verifying A11...  PASSED.

                          DO NOT ABORT!
srmflash      Updating to V6.0-3...  Verifying V6.0-3...  PASSED.

UPD> exit                                                        ❺
```

❹ The **update** command updates the device specified or all devices. In this example, the wildcard indicates that all devices supported by the selected update file will be updated. Typically LFU requests confirmation before updating each console's or device's firmware. The **-all** option eliminates the update confirmation requests.

❺ The **exit** command returns you to the console from which you entered LFU (either SRM or AlphaBIOS).

3.9.5   LFU Commands

The commands summarized in Table 3–2 are used to update system firmware.

Table 3–2   LFU Command Summary

Command	Function
display	Shows the physical configuration of the system.
exit	Terminates the LFU program.
help	Displays the LFU command list.
lfu	Restarts the LFU program.
list	Displays the inventory of update firmware on the selected device.
readme	Lists release notes for the LFU program.
update	Writes new firmware to the module.
verify	Reads the firmware from the module into memory and compares it with the update firmware.

These commands are described in the following pages.

display

The **display** command shows the physical configuration of the system. **Display** is equivalent to issuing the SRM console command **show configuration**. Because it shows the slot for each module, **display** can help you identify the location of a device.

exit

The **exit** command terminates the LFU program, causes system initialization and testing, and returns the system to the console from which LFU was called.

help

The **help** (or **?**) command displays the LFU command list, shown below.

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

lfu

The **lfu** command restarts the LFU program. This command is used when the update files are on a floppy disk. The files for updating both console firmware and I/O firmware are too large to fit on a 1.44 MB disk, so only one type of firmware can be updated at a time. Restarting LFU enables you to specify another update file.

list

The **list** command displays the inventory of update firmware on the CD-ROM, network, or floppy. Only the devices listed at your terminal are supported for firmware updates.

The **list** command shows three pieces of information for each device:

- Current Revision — The revision of the device's current firmware
- Filename — The name of the file used to update that firmware
- Update Revision — The revision of the firmware update image

readme

The **readme** command lists release notes for the LFU program.

update

The **update** command writes new firmware to the module. Then LFU automatically verifies the update by reading the new firmware image from the module into memory and comparing it with the source image.

To update more than one device, you may use a wildcard but not a list. For example, **update k*** updates all devices with names beginning with k, and **update *** updates all devices. When you do not specify a device name, LFU tries to update all devices; it lists the selected devices to update and prompts before devices are updated. (The default is no.) The **-all** option eliminates the update confirmation requests, enabling the update to proceed without operator intervention.

*CAUTION: Never abort an **update** operation. Aborting corrupts the firmware on the module.*

verify

The **verify** command reads the firmware from the module into memory and compares it with the update firmware. If a module already verified successfully when you updated it, but later failed tests, you can use **verify** to tell whether the firmware has become corrupted.

3.10   Hard Disk Partitioning

The recommended hard disk partition on the first hard disk in your system is: partition 1 should be 6 megabytes less than the total size of the drive (this large partition holds the operating system and the application and data files) and partition 2 should be the remaining 6 megabytes (this small partition holds only the few files necessary for your computer to boot).

This arrangement provides two benefits:

- Windows NT requires that a boot partition be formatted with the FAT file system. However, the Windows NT file system (NTFS) provides advantages over FAT, such as additional security and more efficient use of disk space. By keeping the FAT boot partition as small as possible, the maximum amount of space is left available for use as an NTFS partition.

- Most applications install themselves onto drive C by default. Although the program installation drive is usually configurable by the user, many people accept the default of drive C. By making the first partition large, drive C becomes the larger drive. This arrangement makes program installation easier and avoids time-consuming insufficient disk space mistakes.

3.10.1   Hard Disk Error Conditions

Disk Initialization Failed

When you start hard disk setup, if you receive an "Internal error occurred" message, it means that a disk was found, but there was an error in communicating with the disk. The likely conditions that can cause this error are:

- Incompatible or failed disk cables. You may not have the correct cable installed, or the cable might have a broken lead or connector. Try another cable known to be good.

- Disk controller not configured. You may have to run a configuration utility to set up your hard disk controller. Check your controller documentation.

- Disk controller malfunction. Most controllers come with a diagnostic utility to test controller functioning. If a controller error is found, call the manufacturer for a replacement.

- Improper SCSI termination. Many SCSI controllers require that the terminating resistor packs be removed from all drives between the controller and last drive. Only the controller itself, and the last drive connected to the controller (the ends of the "chain"), should have terminating resistor packs left on.

No Hard Disks Found

When you start hard disk setup, if you receive a "No hard drives were found connected to your computer" message, it means that AlphaBIOS could not locate a hard drive. The likely conditions that cause this error are:

- Cable not connected to either the disk or controller. The cable may have worked loose from the connector on the controller or disk drive. Check the cable connections, making sure the cable connectors are fully seated.

- No power to the drive. The power connector may have worked loose from the receptacle on the drive, or the power cable itself may be malfunctioning. Check the cable connections, making sure the cable connectors are fully seated. Try connecting the drive to a different power connector.

- Disk drive malfunction. The disk drive itself may be malfunctioning and not responding to requests from the controller. If this is the problem, then it would appear as though the disk were absent. Replace the drive with a drive known to be good. If the known good drive is correctly detected, then your hard drive is defective. Contact the manufacturer for a replacement.

No Partitions on Disk

If hard disk 0 does not have any partitions defined, then a message will appear when you start hard disk setup, asking if you want to perform an express disk setup. Express disk setup automatically creates the default disk partition arrangement on hard disk 0.

For more information on express setup, see Chapter 5.

3.10.2   System Partitions

To install Windows NT, a system partition must be defined. If no system partition is found, the user is asked to choose from the available FAT partitions.

The purpose of the system partition is twofold. First, it tells the Windows NT installation program where to place the OS Loader and hardware support files. Second, upon subsequent restarts of Windows NT, the system partition definition tells AlphaBIOS where the OSLOADER.EXE file is so it can successfully hand off control to the OS Loader and continue the boot process.

The system partition can be the same partition into which Windows NT is installed, or it can be separate. However, the system partition must be formatted with the FAT file system. Because the default recommended partition arrangement on Alpha calls for the partition into which Windows NT is installed to be an NTFS partition, the system partition will almost always be a small FAT partition on the same disk onto which Windows NT is installed.

3.10.3   How AlphaBIOS Works with System Partitions

If you are installing Windows NT for the first time, AlphaBIOS will determine that a system partition has not been defined when you select **Install Windows NT** in the **AlphaBIOS Setup** screen (see Figure 3–5). When this occurs, AlphaBIOS searches for all FAT partitions on the system. If only one FAT partition exists, AlphaBIOS designates that FAT partition as the system partition and continues with the Windows NT installation. If more than one FAT partition exists on your system, AlphaBIOS displays the list of FAT partitions from which you can choose the system partition. After choosing the system partition, the installation process continues.

Figure 3–5   System Partition Not Defined

```
                          AlphaBIOS Setup


   ┌──────────────────────────────────────────┐
   │ Display System Configuration              │
   │ Upgrade AlphaBIOS                         │
   │ Hard Disk Setup...                        │
   │ CMOS Se ┌───────────No System Partition───────────┐
   │ Install │                                          │
   │ Utiliti │  You must designate a FAT disk partition as a System │
   │ About A │  Partition.  Windows NT will use the System Partition │
   │         │  to store system specific startup files. │
   │         │  Please select a partition as the System Partition. │
   │         │                                          │
   │         │  System Partition: ██Disk 0, Partition 1██▼│
   │         ├──────────────┬───────────────────────────┤
   │  ENTER=Continue   E│Disk 0, Partition 1         │
   │         │           │██Disk 0, Partition 2██      │
   │         │           │Disk 1, Partition 1          │
   │         │           │Disk 1, Partition 2          │
   │         │           └─────────────────────────────┘
   └──────────────────────────────────────────┘
```

PK-0748-96

Operation   **3-49**

3.11   Using the Halt Button

Use the Halt button to halt the DIGITAL UNIX or OpenVMS operating system when it hangs, clear the SRM console password, or force a halt assertion, as described in Section 3.12.

Using Halt to Shut Down the Operating System

You can use the Halt button if the DIGITAL UNIX or OpenVMS operating system hangs. Pressing the Halt button halts the operating system back to the SRM console firmware. From the console, you can use the **crash** command to force a crash dump at the operating system level. See Section 4.11.

The Windows NT operating system does not support halts on this system. Pressing the Halt button during a Windows NT session has no effect.

Using Halt to Clear the Console Password

The SRM console firmware allows you to set a password to prevent unauthorized access to the console. If you forget the password, the Halt button, with the **login** command, lets you clear the password and regain control of the console. Section 4.8 describes the procedure.

3.12   Halt Assertion

A halt assertion allows you to disable automatic boots of the operating system so that you can perform tasks from the SRM console.

Under certain conditions, you might want to force a "halt assertion." A halt assertion differs from a simple halt in that the SRM console "remembers" the halt. The next time you power up, the system ignores the SRM power-up script (nvram) and ignores any environment variables that you have set to cause an automatic boot of the operating system. The SRM console displays this message:

```
Halt assertion detected
NVRAM power-up script not executed
AUTO_ACTION=BOOT/RESTART and OS_TYPE=NT ignored, if applicable
```

Halt assertion is useful for disabling automatic boots of the operating system when you want to perform tasks from the SRM console. It is also useful for disabling the SRM power-up script if you have accidentally inserted a command in the script that will cause a system problem. These conditions are described in the sections "Disabling Autoboot" and "Disabling the SRM Power-Up Script."

You can force a halt assertion using the Halt button, the RCM **halt** command, or the RCM **haltin** command. Observe the following guidelines for forcing a halt assertion.

Halt Assertion with Halt Button or RCM Halt Command

Press the Halt button on the local system (or enter the RCM **halt** command from a remote system) while the system is powering up or the SRM console is running. The system halts at the SRM console, and the halt status is saved. The next time the system powers up, the saved halt status is checked.

*NOTE: Wait 5 seconds after the system begins powering up before pressing the Halt button or remotely entering the RCM **halt** command.*

Halt Assertion with RCM Haltin Command

Enter the RCM **haltin** command at any time except during power-up. For example, enter **haltin** during an operating system session or when the AlphaBIOS console is running.

If you enter the RCM **haltin** command during a DIGITAL UNIX or OpenVMS session, the system halts back to the SRM console, and the halt status is saved. The next time the system powers up, the saved halt status is checked.

If you enter the RCM **haltin** command when Windows NT or AlphaBIOS is running,

the interrupt is ignored. However, you can enter the RCM **haltin** command followed by the RCM **reset** command to force a halt assertion. Upon reset, the system powers up to the SRM console, but the SRM console does not load the AlphaBIOS console.

Clearing a Halt Assertion

Clear a halt assertion as follows:

- If the halt assertion was caused by pressing the Halt button or remotely entering the RCM **halt** command, the console uses the halt assertion once, then clears it.

- If the halt assertion was caused by entering the RCM **haltin** command, enter the RCM **haltout** command or cycle power on the local system.

Disabling Autoboot

The system automatically boots the selected operating system at power-up or reset if the following environment variables are set:

- For DIGITAL UNIX and OpenVMS, the SRM environment variables **os_type**, **auto_action**, **bootdef_dev**, **boot_file**, and **boot_osflags**

- For Windows NT, the SRM **os_type** environment variable and the **Auto Start** selection in the AlphaBIOS **Standard CMOS Setup** screen

You might want to prevent the system from autobooting so you can perform tasks from the SRM console. Use one of the methods described previously to force a halt assertion. When the SRM console prompt is displayed, you can enter commands to configure or test the system. Chapter 4 describes the SRM console commands and environment variables.

Disabling the SRM Power-Up Script

The system has a power-up script (file) named "nvram" that runs every time the system powers up. If you accidentally insert a command in the script that will cause a system problem, disable the script by using one of the methods described previously to force a halt assertion. When the SRM console prompt is displayed, edit the script to delete the offending command. See Section 4.4 for more information on editing the nvram script.

Chapter 4

SRM Console

The SRM console is the command line interface that supports the DIGITAL UNIX and OpenVMS operating systems.   The SRM console is used to bootstrap the operating system, configure and test the system hardware, examine system options for errors, and set or change environment variables.

This chapter describes the SRM commands and environment variables. Sections in this chapter are:

- Invoking the SRM Console
- Command Summary
- Displaying the System Configuration
- Creating a Power-Up Script
- Booting the Operating System
- Configuring the System
- Testing the System
- Making the System Secure
- Stopping and Starting CPUs
- Updating Firmware
- Forcing a System Crash Dump
- Using Environment Variables
- Depositing and Examining Data
- Reading a File
- Initializing the System
- Finding Help
- Switching from SRM to AlphaBIOS Console
- Environment Variable Summary

4.1 Invoking the SRM Console

Invoke the SRM console by shutting down the operating system or by pressing the Halt button on the control panel.

In a system running the DIGITAL UNIX or OpenVMS Alpha operating system, do one of the following steps to invoke the SRM console mode:

- Shut down the operating system according to the procedure described in your operating system documentation. The SRM console prompt, P00>>>, will be displayed.

- Press the Halt button on the control panel. The console prompt, P00>>>, will be displayed.

If you are in AlphaBIOS and want to switch to the SRM console, do one of the following steps:

- Press the Reset button, wait 5 seconds, then press the Halt button. The AlphaBIOS firmware will boot.

- Select either **OpenVMS console (SRM)** or **DIGITAL UNIX console (SRM)** from the **Console Selection** in the **Advanced CMOS Setup** screen. Reset the system to start the SRM console.

After you have performed tasks in the console mode, you must boot the operating system with the **boot** command to go back to the operating mode.

4.2 Command Summary

The SRM (*Alpha System Reference Manual*) console is a command line interface used when the operating system is either DIGITAL UNIX or OpenVMS. Console commands enable the user to examine and modify the system state.

Table 4-1 gives the most commonly used SRM console commands.

Table 4-2 gives the syntax for the console commands.

Table 4-3 gives special characters you can use in console mode.

Table 4-1   Summary of SRM Console Commands

Command	Function
alphabios	Loads and starts the AlphaBIOS console.
boot	Loads and starts the operating system.
clear *envar*	Resets an environment variable to its default value.
clear password	Sets the password to zero.
continue	Resumes program execution.
crash	Forces a crash dump at the operating system level.
deposit	Writes data to the specified address.
edit	Invokes the console line editor on a RAM file or on the nvram file (power-up script).
examine	Displays the contents of a memory location, register, or device.
halt	Halts the specified processor. (Same as **stop**.)
help	Displays information about the specified console command.
initialize	Resets the system.
lfu	Runs the Loadable Firmware Update Utility.
login	Turns off secure mode, enabling access to all SRM console commands during the current session.
man	Displays information about the specified console command.

Continued on next page

Table 4-1 Summary of SRM Console Commands (Continued)

Command	Function
more	Displays a file one screen at a time.
prcache	Utility that initializes and displays status of the PCI NVRAM.
set *envar*	Sets or modifies the value of an environment variable.
set host	Connects to a MSCP DUP server on a DSSI device.
set password	Sets the console password for the first time or changes an existing password.
set secure	Enables secure mode without requiring a restart of the console.
show *envar*	Displays the state of the specified environment variable.
show config	Displays the configuration at the last system initialization.
show cpu	Displays the state of each processor in the system.
show device	Displays a list of controllers and their devices in the system.
show fru	Displays the serial number and revision level of system bus options.
show memory	Displays memory module information.
show network	Displays the state of network devices in the system.
show pal	Displays the version of the privileged architecture library code (PALcode).
show power	Displays information about the power supplies, system fans, CPU fans, and temperature.
show version	Displays the version of the console program.
start	Starts a program that was previously loaded on the processor specified.
stop	Halts the specified processor. (Same as **halt**.)
test	Runs firmware diagnostics for the system.

Table 4-2   Syntax for SRM Console Commands

Parameter	Attribute or Action
Length	Up to 255 characters, not including the terminating carriage return or any characters deleted as the command is entered. A command longer than 80 characters and without the backslash character (see Table 4-3) causes display of an error message.
Case	Upper- or lowercase characters can be used for input. Characters are displayed in the case in which they are entered.
Abbreviation	Only by dropping characters from the end of words. You must enter the minimum number of characters to identify the keyword unambiguously. Abbreviation of environment variables is allowed with the **show** command.
Options	You can use command options, to modify the environment, after the command keyword or after any symbol or number in the command. See individual command descriptions for examples.
Numbers	Most numbers in console commands are in decimal notation. Two exceptions, both of which use hexadecimal notation, are addresses and numbers used in the **deposit** command. The default radix can be overridden by inserting %d before numbers you want to express in decimal, %b before binary, %o before octal, or %x before hexadecimal. Register names (for example, R0) are not considered numbers and use decimal notation.
No characters	A command line with no characters is a null command. The console program takes no action and does not issue an error message; it returns the console prompt. The console supports command line recall and editing.
Spaces or tabs	Multiple adjacent spaces and tabs are compressed and treated as a single space. The console program ignores leading and trailing spaces.

Table 4-3 Special Characters for SRM Console

Character	Function
Return or Enter	Terminates a command line. No action is taken on a command until it is terminated. If no characters are entered and this key is pressed, the console just redisplays the prompt.
Backslash (\)	Continues a command on the next line. Must be the last character on the line to be continued.
Delete	Deletes the previous character.
Help	By itself, displays first-level help. When pressed after part of a command, displays options available.
Ctrl/A or F14	Toggles between insert and overstrike modes. The default is overstrike.
Ctrl/B or up-arrow	Recalls previous command or commands. The last 16 commands are stored in the recall buffer.
Ctrl/C or Ctrl/P	Terminates the process that is running. Clears Ctrl/S; resumes output suspended by Ctrl/O. When entered as part of a command line, deletes the current line. Ctrl/C has no effect as part of a binary data stream.
Ctrl/D or left-arrow	Moves the cursor left one position.
Ctrl/E	Moves the cursor to the end of the line.
Ctrl/F or right-arrow	Moves the cursor right one position.
Ctrl/H or Backspace or F12	Moves the cursor to the beginning of the line.
Ctrl/J	Deletes the previous word.
Ctrl/O	Stops output to the console terminal for the current command. Toggles between enable and disable. The output can be reenabled by other means as well: when the console prompts for a command, issues an error message, or enters program mode, or when Ctrl/P is entered.
Ctrl/Q	Resumes output to the console terminal that was suspended by Ctrl/S.

Table 4-3   Special Characters for SRM Console (Continued)

Character	Function
Ctrl/R	Redisplays the current line. Deleted characters are omitted. This command is useful for hardcopy terminals.
Ctrl/S	Suspends output to the console terminal until Ctrl/Q is entered. Cleared by Ctrl/C.
Ctrl/U	Deletes the current line.
*	Wildcarding for commands such as **show**.
" "	Double quotes enable you to denote a string for environment variable assignment.
#	Specifies that all text between it and the end of the line is a comment. Control characters are not considered part of a comment.

4.3 Displaying the System Configuration

Several commands are used to display the system configuration: show config, show cpu, show device, show fru, show memory, show network, show pal, show power, and show version.

Example 4-1 Show Config Command

```
P00>>> show config
              Digital Equipment Corporation
                  AlphaServer 1200

Console V5.0-2  OpenVMS PALcode V1.19-12, DIGITAL UNIX PALcode V1.21-20

Module                            Type     Rev     Name
System Motherboard                0        0003    mthrbrd0
Memory  256 MB DIMM               0        0000    mem0
CPU (4MB Cache)                   3        0001    cpu0
CPU (4MB Cache)                   3        0001    cpu1
Bridge (IOD0/IOD1)                600      0032    iod0/iod1
PCI Motherboard                   a        0003    saddle0

Bus 0 iod0 (PCI0)
Slot   Option Name               Type     Rev     Name
1      PCEB                       4828086  0005    pceb0
3      S3 Trio64/Trio32           88115333 0000    vga0
4      DECchip 21140-AA           91011    0012    tulip1

Bus 1 pceb0 (EISA Bridge connected to iod0, slot1)
Slot   Option Name               Type     Rev     Name

Bus 0 iod1 (PCI1)
Slot   Option Name               Type     Rev     Name
1      NCR 53C810                 11000    0002    ncr0
2      DECchip 21040-AA           21011    0023    tulip0
3      ATI Mach64                 43541002 0009    vga1
4      PCI-PCI Bridge             11011    0002    pcb0

Bus 2 pcb0 (PCI-PCI Bridge connected to iod1, slot 4)
Slot   Option Name               Type     Rev     Name
0      Qlogic ISP1020             10201077 0002    isp0
P00>>>
```

The **show config** command displays a list of devices found on the system bus and I/O buses. This is the configuration at the most recent initialization. The syntax is:

show config

Example 4-2   Show CPU Command

```
P00>>> show cpu

Primary CPU:        00
Active CPUs:        00     01
Configured CPUs:    00     01
SROM Revision:      V3.0   V3.0

P00>>>
```

The **show cpu** command displays the status of each CPU. The syntax is:

show cpu

Example 4-3 Show Device Command

```
P00>>> show dev
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1    SCSI Bus ID 7
dka500.5.0.1.1     Dka500                        RRD46   0557
polling isp0 (QLogic ISP1020) slot 0, bus 2 PCI, hose 1    SCSI Bus ID 7
dkb0.0.0.2000.1    DKb0                          RZ29B   0007
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0    DVA0                          RX23
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1       08-00-2B-E6-4B-CC  BNC
polling tulip1 (DECchip 21140-AA) slot 4, bus 0 PCI, hose 0
ewb0.0.0.4.0       00-00-F8-02-23-74  AUI
P00>>>
```

The **show device** command displays status for devices and controllers in the system:
SCSI and MSCP devices, the internal floppy drive, and the network. The syntax is:

show device [*controller_name*]

controller_name   The controller name or abbreviation. When abbreviations or
                     wildcards are used, all controllers that match the type are
                     displayed. If no name is given, the display is a list of all devices
                     and controllers in the system.

An example of a device name is **dka500.5.0.1.1**. Table 4-4 shows the interpretation
of this device name.

Table 4-4 Device Naming Convention

	Category	Description
dk	Driver ID	Two-letter designator of port or class driver

dk	SCSI disk	fw	FDDI device
dr	RAID set device	mk	SCSI tape
du	DSSI disk	mu	DSSI tape
dv	Diskette drive	pk	SCSI port
ew	Ethernet port	pu	DSSI port

	Category	Description
a	Storage adapter ID	One-letter designator of storage adapter (a, b, c...).
500	Device unit number	Unique number (MSCP unit number). SCSI unit numbers are forced to 100 X node ID.
5	Bus node number	Bus node ID.
0	Channel number	Used for multi-channel devices.
1	Logical slot number	Corresponds to PCI slot number, as shown in Table 4-5.
1	Hose number	0 — PCI 0 1 — PCI 1

Table 4-5 PCI Slot Numbers

Slot	PCI 0	PCI 1	EISA/ISA
1	EISA bridge	Embedded SCSI CD-ROM	EISA or ISA device
2	PCI device	PCI device	
3	PCI device	PCI device	
4	PCI device	PCI device	

Example 4-4   Show FRU Command

```
P00>>> show fru
                        Digital Equipment Corporation
                              AlphaServer 1200

Console V5.0-2  OpenVMS PALcode V1.19-12, Digital UNIX PALcode V1.21-20

Module                 Part #    Type    Rev     Name      Serial #
System Motherboard     25147-01  0       0000    mthrbrd0  NI72000047
Memory   64 MB DIMM    N/A       0       0000    mem0      N/A
Memory   64 MB DIMM    N/A       0       0000    mem1      N/A
Memory   64 MB DIMM    N/A       0       0000    mem2      N/A
Memory   64 MB DIMM    N/A       0       0000    mem3      N/A
CPU (4MB Cache)        B3004-DA  3       0000    cpu0      KA705TRVNS
Bridge (IOD0/IOD1)     25147-01  600     0032    iod0/iod1 NI72000047
PCI Motherboard        25147-01  a       0003    saddle0   NI72000047


Bus 0  iod0 (PCI0)
Slot   Option Name            Type      Rev     Name
1      PCEB                   4828086   0005    pceb0
2      S3 Trio64/Trio32       88115333  0054    vga0
3      DECchip 21041-AA       141011    0011    tulip0

Bus 1  pceb0 (EISA Bridge connected to iod0, slot 1)
Slot   Option Name            Type      Rev     Name

Bus 0  iod1 (PCI1)
Slot   Option Name            Type      Rev     Name
1      NCR 53C810             11000     0002    ncr0
4      QLogic ISP1020         10201077  0005    isp0
```

The **show fru** command displays information for CPU, memory, and PCI. This
information is based on data in the serial control bus EPROM. The syntax is:

show fru

Example 4-5 Show Memory Command

```
P00>>> show memory

Slot    Type       MB     Base
----    ----       ----   --------
0       DIMM       64     0
1       DIMM       64     20000000
2       DIMM       64     40000000
3       DIMM       64     60000000

Total              256
P00>>>
```

The **show memory** command displays information about each memory pair: slot
number, type (DIMM), size of the memory pair in megabytes, and the starting
address of the pair. The syntax is:

show memory

Example 4-6 Show Network Command

```
P00>>> show net
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1        08-00-2B-E6-4B-CC      BNC
polling tulip1 (DECchip 21140-AA) slot 4, bus 0 PCI, hose 0
ewb0.0.0.4.0        00-00-F8-02-23-74      AUI
P00>>>
```

The **show network** command displays the assigned network device names and other
information about network (FDDI and Ethernet) cards. The syntax is:

show network

Example 4-7 Show PAL Command

```
P00>>> show pal
pal  OpenVMS PALcode V1.19-12, DIGITAL UNIX PALcode V1.21-20
P00>>>
```

The **show pal** command displays the versions of DIGITAL UNIX and OpenVMS
PALcode. PALcode is the Alpha Privileged Architecture Library code, written to
support Alpha processors. It implements architecturally defined processor behavior.
The syntax is:

show pal

Example 4-8 Show Power Command

```
P00>>> show power

                     Status
Power Supply 0        good
Power Supply 1        good
System Fans           good
CPU Fans              good
Temperature           good

Current ambient temperature is 23 degrees C
System shutdown temperature is set to 55 degrees C

The system was last reset via a system software reset

0 Environmental events are logged in nvram
Do you want to view the events? (Y/<N>) y

Total Environmental Events: 5 (5 logged)

1   JUL 25  1:17  Temperature, Fans, Power Supplies Normal
2   Jul 29 23:44  Temperature, Fans, Power Supplies Normal
3   Jul 29 23:51  Temperature, Fans, Power Supplies Normal
4   Jul 30  0:18  System Fan 0 Failure
5   Jul 30 19:22  Temperature, Fans, Power Supplies Normal

Do you want to clear all events from nvram? (Y/<N>) y
P00>>>
```

The **show power** command displays status information about the power supplies, system fans, CPU fans, and temperature. This command is useful for displaying the error state of a DIGITAL UNIX or OpenVMS system that shuts down because of a fan, temperature, or power supply failure. If the system can be restarted, use this command. (If it cannot, use the RCM **status** command. See Chapter 6.)

The syntax is:

show power

Example 4-9 Show Version Command

```
P00>>> show version
version               V5.0-2 20-OCT-1997 13:59:28
P00>>>
```

The **show version** command displays the version of the SRM console program that is installed on the system. The syntax is:

show version

4.4 Creating a Power-Up Script

The system comes with a special nonvolatile file named "nvram" that is stored in EEROM. Nvram is a user-created power-up script (set of commands) that is always invoked during the power-up sequence. Use the SRM edit command to create or alter the nvram script.

Example 4-10   Editing the nvram Script

```
P00>>> edit nvram            #Modify user power-up script, nvram
editing 'nvram'
0 bytes read in
*10 set ewa0_protocols bootp
*list                         #List current file with line numbers
10  set ewa0_protocols bootp
*exit                         #Close file and save changes
27 bytes written out to nvram
P00>>> nvram                  #Execute the script.
```

This example shows how to modify the user-created power-up script, "nvram." The pound sign (#) indicates explanatory comments. In this example the script is edited to include a command that allows you to boot the DIGITAL UNIX operating system over the network.

Example 4-11   Clearing the nvram Script

```
P00>>> edit nvram
editing 'nvram'
20 bytes read in
*10
*exit
0 bytes written out to nvram
P00>>>
```

To clear the script, enter line numbers without any text. This deletes the lines.

Editing the Nvram Script

You can create an nvram script to include any commands you want the system to execute at power-up.

You create and edit the nvram script using the SRM **edit** command. With **edit**, lines may be added, overwritten, or deleted.

The syntax is:

edit *file*

file is the name of the file to be edited.

The editing commands are:

help	Displays the brief help file.
list	Lists the current file prefixed with line numbers.
renumber	Renumbers the lines of the file in increments of 10.
exit	Leaves the editor and closes the file, saving all changes.
quit	Leaves the editor and closes the file without saving changes.
nn	Deletes line number *nn*.
nn text	Adds or overwrites line number *nn* with *text*.

*NOTE: It is possible to disable the system by editing the nvram script. For example, if you include the **initialize** command in the script, the system will go into an endless loop. To fix this, press the Halt button while the system is powering up. You can then edit the script to delete the offending command.*

4.5 Booting the Operating System

The boot command is used to boot the operating system.

Example 4-12   Boot Command

```
P00>>> boot dkb0
 SROM V3.0 on cpu0
 SROM V3.0 on cpu1
XSROM V5.0 on cpu0
XSROM V5.0 on cpu1
BCache testing complete on cpu0
BCache testing complete on cpu1
mem_pair0 - 64 MB
mem_pair1 - 64 MB
mem_pair2 - 64 MB
mem_pair3 - 64 MB
20..20..21..21..23..
please wait 9 seconds for T24 to complete
24..24..
Memory testing complete on cpu0
Memory testing complete on cpu1
starting console on CPU 0
sizing memory
   0    64 MB DIMM
   1    64 MB DIMM
   2    64 MB DIMM
   3    64 MB DIMM
starting console on CPU 1
probing IOD1 hose 1
   bus 0 slot 1 - NCR 53C810
   bus 0 slot 4 - Qlogic ISP1020
probing IOD0 hose 0
   bus 0 slot 1 - PCEB
probing EISA Bridge, bus 1
bus 0 slot 2 - S3 Trio64/Trio32
bus 0 slot 3 - DECchip 21040-AA

configuring I/O adapters...
ncr0, hose 1, bus 0, slot 1
isp0, hose 1, bus 0, slot 4
floppy0, hose 0, bus 1, slot 0
tulip1, hose 0, bus 0, slot 3
System temperature is 20 degrees C
AlphaServer 1200 Console V5.0-2, 19-NOV-1997 13:57:07

CPU 0 booting
```

```
(boot dkb0.0.0.4.1 -flags a)
.
.
.
setting affinity to the primary CPU
jumping to bootstrap code

     Digital UNIX boot –Fri Nov 22 21:24:46  EST 1997
```

The **boot** command initializes the processor, loads a program image from the specified boot device, and transfers control to that image. The syntax is:

boot [-file *filename***] [-flags [***value***]] [-halt] [-protocols** *enet_protocol***] [***boot_dev***]**

-file *filename*	The boot file.
-flags [*value*]	Specifies additional information to the loaded image or operating system. In DIGITAL UNIX, specifies boot flags. In OpenVMS, specifies system root number and boot flags. This qualifier overrides the setting of the **boot_osflags** environment variable. See the **boot_osflags** environment variable for a list of settings and their meanings.
-halt	Forces the bootstrap operation to halt and invoke the console program once the bootstrap image is loaded and page tables and other data structures are set up. Console device drivers are not shut down. Transfer control to the image by entering the **continue** command.
-protocols *enet_protocol*	Either **mop** (default) or **bootp**. This qualifier overrides the setting of the **ew*0_protocols** environment variable.
boot_dev	A device path or list of devices from which the console program attempts to boot, or a saved boot specification in the form of an environment variable. This qualifier overrides the setting of the **bootdef_dev** environment variable. Use the **bootdef_dev** environment variable to define the default boot device string.

4.6 Configuring the System

4.6.1 Configuring DSSI

The set host command is used for system configuration when a DSSI device is in the system.

Example 4-13 Set Host Command

```
P00>>> show device kfpsa0
   dud0.0.0.3.0                  R2YQYA$DIA0                RF72
P00>>> set mode diag
P00>>> config kfpsa*
P00>>> set host -dup dud0
    starting DIRECT on pud0.0.0.3.0 (R2YQYA)

    Copyright (C) 1997  Digital Equipment Corporation
    PRFMON V1.0  D  2-NOV-1995 10:30:58
    DKCOPY V1.0  D  2-NOV-1995 10:30:58
    DRVEXR V2.0  D  2-NOV-1995 10:30:58
    DRVTST V2.0  D  2-NOV-1995 10:30:58
    HISTRY V1.0  D  2-NOV-1995 10:30:58
    DIRECT V1.0  D  2-NOV-1995 10:30:58
    ERASE  V2.0  D  2-NOV-1995 10:30:58
    VERIFY V1.0  D  2-NOV-1995 10:30:58
    DKUTIL V1.0  D  2-NOV-1995 10:30:58
    PARAMS V2.0  D  2-NOV-1995 10:30:58
    Total of 10 programs.
    Task?
P00>>> set host -dup -task params dud0
    starting PARAMS on pud0.0.0.3.0 (R2YQYA)

    Copyright (C) 1995  Digital Equipment Corporation

    PARAMS> show allclass

    Parameter    Current       Default      Type    Radix
    ---------  -------------  ------------  -------  ------
    ALLCLASS              0            0    Byte     Dec  B

    PARAMS> exit
    Exiting...
P00>>>
```

The **set host** command connects the console program to another DUP server on an MSCP DSSI device. You can use the DUP protocol to examine and modify parameters of a DSSI device. The syntax is:

set host -dup [**-task** *task_name*] *device*

-dup	Connect to an MSCP DUP server. The DUP service may be used to examine and modify parameters of a DSSI device.
-task *task_name*	Specifies the DUP utility to invoke. If **-task** is not specified, a directory of utilities displays.
device	The device to attach to the console program.

4.6.2 Configuring a PCI NVRAM Module

The prcache command is used for system configuration when a PCI NVRAM module is in the system. The command is used only with DIGITAL UNIX systems.

Example 4-14   Prcache Command

```
P00>>> prcache -f
PCI NVRAM Disk Cache: passed
Size:   4MB
PCI Memory Address: 40000000
System ID: 12000000
State:  - not valid
Battery Status: good (Charging)
Battery Disconnect Circuit Status: enabled

P00>>> prcache -z
This command will zero the PCI NVRAM Disk Cache
Do you really want to continue [Y/N] ? : y
clearing disk cache
P00>>>
```

The **prcache** command is used to check PCI NVRAM configuration information and battery status, to clear data from the NVRAM module, and to set the date of the next battery replacement. This command is used only with DIGITAL UNIX systems. The syntax is:

prcache -{f,z,b}

-f   Checks configuration information and battery status.

-z   Clears valid data; writes zeros to memory.

-b   Sets the date (month and year) for battery replacement.

4.7 Testing the System

The test command runs firmware diagnostics for components of the system drawer. This command is used to test Windows NT systems as well as DIGITAL UNIX and OpenVMS systems.

Example 4-15 Test Command

```
P00>>> test
Console is in diagnostic mode
System test, runtime 1200 seconds

Type ^C if you wish to abort testing once it has started

Configuring adapters..
polling ncr0 (NCR 53C810) slot 1, bus 0 PCI, hose 1   SCSI Bus ID 7
dka500.5.0.1.1     Dka500                  RRD46   0557
polling isp0 (QLogic ISP1020) slot 0, bus 2 PCI, hose 1   SCSI Bus ID 7
dkb0.0.0.2000.1    DKb0                    RZ29B   0007
polling floppy0 (FLOPPY) PCEB - XBUS hose 0
dva0.0.0.1000.0    DVA0                    RX23
polling tulip0 (DECchip 21040-AA) slot 2, bus 0 PCI, hose 1
ewa0.0.0.2.1       08-00-2B-E6-4B-CC  BNC
polling tulip1 (DECchip 21140-AA) slot 4, bus 0 PCI, hose 0
ewb0.0.0.4.0       00-00-F8-02-23-74  AUI

Testing Ethernet device(s)
External loopback failed on ewb0 - check network connection

Testing VGA (alphanumeric mode only)

Starting phase 1 background cache/memory tests, affinity to all CPUs..
Starting phase 1 processor/cache thrasher on CPU0..
```

Continued on next page

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

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

ID	Program	Device	Pass	Hard/Soft		Bytes Written	Bytes Read
00001a64	memtest	memory	1	0	0	125829120	125829120
00001acf	memtest	memory	2	0	0	67108800	67108800
00001aeb	memtest	memory	2	0	0	67108800	67108800
00001afd	memtest	memory	2	0	0	67108800	67108800
00001afe	memtest	memory	2	0	0	67108800	67108800
00001b3d	memtest	memory	5	0	0	56622816	56622816
00001b47	memtest	memory	2	0	0	48758688	48758688
00001c1c	exer_kid	dkb0.0.0.200	0	0	0	0	10305536
00001c44	exer_kid	dva0.0.0.100	0	0	0	0	0

ID	Program	Device	Pass	Hard/Soft		Bytes Written	Bytes Read
00001a64	memtest	memory	1	0	0	247463936	247463936
00001acf	memtest	memory	4	0	0	201326400	201326400
00001aeb	memtest	memory	4	0	0	201326400	201326400
00001afd	memtest	memory	4	0	0	201326400	201326400
00001afe	memtest	memory	4	0	0	201326400	201326400
00001b3d	memtest	memory	17	0	0	201325568	201325568
00001b47	memtest	memory	7	0	0	195034752	195034752
00001c1c	exer_kid	dkb0.0.0.200	0	0	0	0	34701312
00001c44	exer_kid	dva0.0.0.100	0	0	0	0	491520

```
Testing aborted. Shutting down tests.
Please wait..


System test complete

^C
P00>>>
```

The **test** command runs console-based exercisers for devices in the system.

Testing can be aborted by Ctrl/C.

*NOTE: If you are running the Microsoft Windows NT operating system, switch from AlphaBIOS to the SRM console in order to enter the **test** command. From the AlphaBIOS console, press the Reset button, wait 5 seconds, then press the Halt button. Or, select **DIGITAL UNIX Console (SRM)** or **OpenVMS Console (SRM)** from the **Advanced CMOS Setup** screen and reset the system.*

The syntax is:

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

-t *time*	Specifies the run time in seconds. The default for system test is 1200 seconds (20 minutes).
-q	Disables the display of status messages as exerciser processes are started and stopped during testing. Sets the environment variable **d_verbose** to zero.
option	Either **cpu***n*, **mem***n*, or **pci***n*, where *n* is 0, 1, or *. If nothing is specified, the entire system is tested.

4.8 Making the System Secure

Placing the console in secure mode ensures that unauthorized persons cannot gain access to the system. The commands for console security are set password, clear password, and set secure. The login command turns off security features during the current console session.

Example 4-16   Set Password Command

```
P00>>> set password
Please enter the password:              # Password is not
                                        # echoed.
Please enter the password again:        # Validation is not
P00>>>                                  # echoed.

P00>>> set password                     # Changing a password.
Please enter the password:
Please enter the password again:
Now enter the old password:
P00>>>

P00>>> set password                     # Password entered is
Please enter the password:              # too short.
Password length must be between 15 and 30 characters
P00>>>
```

The **set password** command sets the console password for the first time or changes an existing password. It is necessary to set the password only if the system is going to operate in secure mode.

If a password has not been set and the **set password** command is issued, the console prompts for a password and verification.

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 unchanged if the validation password entered does not match the existing password in the NVRAM.

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

The syntax is:

set password

Example 4-17   Set Secure Command

```
P00>>> set secure              # In this example a password
                               # has been set.
Console is secure. Please login.
P00>>> b dkb0
Console is secure - parameters are not allowed.
P00>>> login
Please enter the password:    # Password is not echoed.
P00>>> b dkb0
(boot dkb0.0.0.3.1)
.
.

P00>>> set secure              # Password has not been set.
Secure not set. Please set the password.
P00>>>
```

The **set secure** command enables secure mode without requiring a restart of the console. If the password has been set, the console will be secured and only a small subset of commands can be performed. If a password has not been set, you are prompted to do so.

The syntax is:

set secure

Secure Mode Functions

When the console is in secure mode, the only commands recognized are **boot**, **login**, **continue**, and **start**.

The **boot** command does not accept command line parameters in secure mode. The console boots using the environment variables stored in NVRAM (**boot_file**, **bootdef_dev**, **boot_flags**). After a successful boot, the console is secured if there is a valid password.

The **start** and **continue** commands are valid on a secure console. After either command is executed, the console is secured if there is a valid password. This prevents an intruder from accessing the system.

Example 4-18 Login Command

```
P00>>> login                     # System is not in secure
                                  # mode.
Secure not set. Please set the password.
P00>>>

P00>>> login                     # System is in secure mode.
Please enter the password:       # Password is not echoed.
P00>>>

P00>>> login                     # System is in secure mode.
Please enter the password:       # Incorrect password is
                                  # entered.
Invalid password
P00>>>
```

The **login** command turns off the security features, enabling access to all SRM console commands during the current session. The system automatically returns to secure mode as soon as the **boot**, **continue**, or **start** command is entered or when the system is initialized.

When the **login** command is entered, the user is prompted for the current system password. If a password has not been set, a message is displayed indicating that there is no password in NVRAM. If a password has been set, this prompt is displayed:

```
Please enter the password:
```

If the password entered matches the password in NVRAM when the prompt is redisplayed, the console is no longer in secure mode and all console commands can be performed.

*NOTE: If you enter the **login** command when a halt assertion exists, the command fails, even if you enter the correct password. See Section 3.12 for information on halt assertion.*

If you have forgotten the current password, clear the password as follows:

- From the Local Console Terminal

1. Enter the **login** command:

   ```
   P00>>> login
   ```

2. At the `Enter Password:` prompt, press the Halt button, then press the Return key.

   The password is now cleared and the console cannot be put into secure mode unless a new password is set.

- From the RCM

1. Enter the **login** command:
   ```
   P00>>> login
   ```

2. At the `Enter Password:` prompt, enter the RCM escape sequence.

3. At the RCM>>> prompt, enter the **halt** command and then the **quit** command:

   ```
   RCM>>> halt
   RCM>>> quit
   ```

4. At the SRM console, clear the password:

   P00>>> clear password

Example 4-19 Clear Password Command

```
P00>>> clear password
Please enter the password:    # Password is not echoed.
Password successfully cleared.
P00>>>

P00>>> clear password
Please enter the password:    # Invalid password entered.
Console is secure
P00>>>
```

The **clear password** command clears the **password** environment variable, setting it to zero. This command is used when you want access to all the SRM console commands, but the system is in secure mode. In order to use **clear password**, you must know the current password.

To clear the password without knowing the current password, use the **login** command, with the Halt button or RCM **halt** command, as described in the section on the **login** command.

4.9 Stopping and Starting CPUs

The start, halt, and continue commands are used to stop and start a program on the specified CPU.

Example 4-20   Start, Halt, and Continue Commands

```
P00>>> start &p1

P00>>> halt 1

halted CPU 1

halt code = 1
operator initiated halt
PC = ffffffff8007cc68
P00>>> continue &p1

continuing CPU 1
```

start

The **start** command starts a program that was previously loaded on the processor specified. If none is specified, the program is started on the primary processor. The syntax is:

start [**&p**n] [*address*]

&pn Specifies the processor. n is 0 or 1.

address   The starting address of the program.

halt (or stop)

The **halt** (or **stop**) command stops program execution on a secondary CPU that is still running a booted program. The syntax is:

halt (or **stop**) *processor_number*

processor_number is the logical CPU number displayed by the **show cpu** command.

continue

The **continue** command resumes program execution on the specified processor or on the primary processor if none is specified. The processor begins executing instructions at the address that is currently in the program counter (PC). The processor is not initialized.

The **continue** command is valid only if you have not disturbed the system state and if you halted the system by pressing the Halt button on the control panel or, for OpenVMS systems only, by entering Ctrl/P on the console terminal. The syntax is:

continue [**&p**n] [*address*]

&pn Specifies the processor. n is 0 or 1.

address   The starting address of the program.

*NOTE: Some console commands, for example, **boot**, can alter the machine state so that program mode cannot be successfully resumed (unless you include **-halt** in the **boot** command). If program mode cannot be resumed, reboot the operating system.*

*Other commands that alter machine state are **lfu**, **show device**, **show network**, **test**, and **start**.*

4.10 Updating Firmware

The lfu command is used to update firmware from the SRM console prompt.

Example 4-21 Lfu Command

```
P00>>> lfu

        ***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: cda0

Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS1200FW]: AS1200CP

Copying AS1200CP from DKA500.5.0.1.1 .
Copying [as1200]TCREADME from DKA500.5.0.1.1 .
Copying [as1200]TCSRMROM from DKA500.5.0.1.1 ........................
Copying [as1200]TCARCROM from DKA500.5.0.1.1 .............

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

UPD> list

Device          Current Revision   Filename          Update
Revision
AlphaBIOS       V5.32-0            arcrom            V6.40-1
srmflash        V5.0-2            srmrom            V6.0-3

UPD> update *
```

Continued on next page

SRM Console   **4-33**

Example 4-21 Lfu Command (Continued)

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

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

                         DO NOT ABORT!
AlphaBIOS          Updating to V6.40-1...  Verifying V6.40-1...  PASSED.

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

                         DO NOT ABORT!
srmflash           Updating to V6.0-3...  Verifying V6.0-3...  PASSED.

UPD> exit
```

The **lfu** command starts the Loadable Firmware Update (LFU) Utility. The syntax is:

lfu

NOTE: If the system has been shut down from a booted program (most commonly, the operating system) or in some other way halted back to the SRM console, the system must be reset before running LFU.

See Chapter 3 for more information about LFU.

4.11 Forcing a System Crash Dump

The crash command forces a crash dump to the selected device on DIGITAL UNIX and OpenVMS systems.

Example 4-22   Crash Command

```
P00>>> crash

CPU 0 restarting

DUMP: 401408 blocks available for dumping.
DUMP: 38535 required for a partial dump.
DUMP: 0x805001 is the primary swap with 401407, start our
last 38534
     : of dump at 362873, going to end (real end is one
more, for header)
DUMP.prom: dev SCSI 1 3 0 4 400 0 0, block 131072
DUMP: Header to 0x805001 at 401407 (0x61fff)
DUMP.prom: dev SCSI 1 3 0 4 400 0 0, block 131072
DUMP: Dump to 0x805001: ..................: End 0x805001
DUMP.prom: dev SCSI 1 3 0 4 400 0 0, block 131072
DUMP: Header to 0x805001 at 401407 (0x61fff)
succeeded

halted CPU 0

halt code = 5
HALT instruction executed
PC = fffffc00004e2d64
P00>>>
```

The **crash** command forces a crash dump at the operating system level (DIGITAL UNIX or OpenVMS). This command is used when an error has caused the system to hang and can be halted with the Halt button or the RCM **halt** command. This command restarts the operating system and forces a crash dump to the selected device. The syntax is:

crash *[device]*

device is the name of the device to which the crash dump is written.

4.12 Using 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. You issue an init command to activate a new environment variable.

Example 4-23 Set *envar* and Show *envar* Commands

```
P00>>> show console
console                   graphics
P00>>> set console serial
P00>>> show console
console                   serial
P00>>> init
```

Environment variables are set or changed with the **set *envar*** command and set to default values with the **set -default *envar*** command. Their values are viewed with the **show *envar*** command. User-defined nonvolatile environment variables are created with the **edit** command described in Section 4.4. Section 4.18 describes the environment variables in detail.

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. Environ - ment variables are used to pass configuration information between the console and the operating system. The setting of these variables determines how the system powers up, boots the operating system, and operates. The syntax is:

set [**-default**] *envar value*

-default   Restores an environment variable to its default setting.

envar   The name of the environment variable to be modified.

value   The new value of the environment variable.

Whenever you modify the value of any of the following environment variables, the new value takes effect only after you reset the system by pressing the Reset button or issuing the **initialize** command:

> **console**
> **kbd_hardware_type**
> **language**
> **ocp_text**
> **os_type**

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.

Example 4–24   Creating a User-Defined Environment Variable

```
P00>>> edit nvram
editing 'nvram'
0 bytes read in
*10  set mopv3_boot 1
*exit
17 bytes written out to nvram

P00>>>
```

In Example 4–24 the nvram script is edited so that an environment variable called
"mop3_boot" is created and set to 1 on each power-up. By default, MOP boots send
four MOP V4 requests before defaulting to MOP V3. This user-created environment
variable forces the SRM console to bypass MOP V4 requests. This speeds up MOP
booting on networks with MOP V3 software.

4.13 Depositing and Examining Data

The deposit command stores data in a specified location. The examine command displays the contents of a memory location, a register, or a device.

Example 4-25 Deposit Command

```
P00>>> dep -b -n 1ff pmem:0 0 # Clear first 512 bytes of
                               # physical memory.

P00>>> d -l -n 3 vmem:1234 5  # Deposit 5 into four
                               # longwords starting at
                               # virtual memory address
                               # 1234.

P00>>> d -n 8 r0 ffffffff     # Load GPRs R0 through R8
                               # with -1.

P00>>> d -l -n 10 -s 200 pmem:0 8
                               # Deposit 8 in the first
                               # longword of the first 17
                               # pages in physical memory.

P00>>> d -l pmem:0 0          # Deposit 0 to physical
                               # memory address 0.

P00>>> d + ff                 # Deposit FF to physical
                               # memory address 4.

P00>>> d scbb 820000          # Deposit 820000 to SCBB.
```

Example 4-26 Examine Command

```
P00>>> examine pc             # Examine the program
                               # counter.
PC psr: 0 (    PC)  0000000000001170

P00>>> examine sp             # Examine the stack pointer.
gpr:   F0 (   R30)  0000000000072A60

P00>>> e -n 6 r4              # Examine register R4 and
                               # the next six registers.
```

Continued on next page

Example 4-26   Examine Command (Continued)

```
grp: 20 (     R4) 0000000000005000
grp: 28 (     R5) 000000000FFFE000
grp: 30 (     R6) 00000003F8000C00
grp: 38 (     R7) 0000000053F761AE
grp: 40 (     R8) 0000010000000000
grp: 48 (     R9) 00000003F7800100
grp: 50 (    R10) 00000000000C7FFC

P00>>> examine pmem:400EC      # Examine physical memory.
pmem:   400EC A49D0078A47D0070
```

deposit

The **deposit** command stores data in the location specified. If no options are given with a **deposit** command, 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 response. If the data is smaller than the data size, the higher order bits are filled with zeros. The syntax is shown below.

examine

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

If no options are given with an **examine** command, 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.

The syntax for both commands is:

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.
*space***:**	Device name (or address space) of the device to access.
address	Offset within a device to which data is deposited. Can be:

	dev_name	A device name.
	fpr- *name*	The floating-point register set; *name* is F0 to F31.
	gpr- *name*	The general register set; *name* is R0 to R31.
	ipr- *name*	The internal processor registers.
	pt- *name*	The PALtemp register set; *name* is PT0 to PT23.
	pmem	Physical memory (default).
	vmem	Virtual memory.

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.

4.14 Reading a File

The more command displays a file one screen at a time.

Example 4-27 More Command

```
P00>>> more el          #Display the contents of the
                        #SRM console's event log one
                        # screen at a time.

P00>>> help * | more    #Display the contents of online
                        #help one screen at a time.
```

The **more** command is similar to that used in the MS-DOS and UNIX operating systems. It is useful for displaying output that scrolls too quickly to be viewed. For example, when you power up the system, the system startup messages scroll, and the messages are logged to an event log. When the P00>>> prompt displays, you can use the **more** command to display the contents of the event log file. The syntax is:

more [*file*...]

file is the name of the file to be displayed.

4.15 Initializing the System

The initialize command resets the system and executes the power-up tests.

Example 4-28   Initialize Command

```
P00>>> initialize
Initializing. . .

 SROM V3.0 on cpu0
 SROM V3.0 on cpu1
XSROM V5.0 on cpu1
XSROM V5.0 on cpu0
BCache testing complete on cpu1
BCache testing complete on cpu0
mem_pair0 - 64 MB
20..20..21..21..23..
please wait 9 seconds for T24 to complete
24..24..
Memory testing complete on cpu1
Memory testing complete on cpu0
starting console on CPU 0
sizing memory
   0     64 MB DIMM
starting console on CPU 1
probing IOD1 hose 1
  bus 0 slot 1 - NCR 53C810
probing IOD0 hose 0
  bus 0 slot 1 - PCEB
    probing EISA Bridge, bus 1
  bus 0 slot 2 - S3 Trio64/Trio32
  bus 0 slot 3 - DECchip 21040-AA
  bus 0 slot 5 - NCR 53C810
configuring I/O adapters...
  ncr0, hose 1, bus 0, slot 1
  floppy0, hose 0, bus 1, slot 0
  tulip0, hose 0, bus 0, slot 3
  ncr1, hose 0, bus 0, slot 5
System temperature is 23 degrees C
AlphaServer 1200 Console V5.0-2, 19-NOV-1997 13:57:07
P00>>>
```

The **initialize** command resets the system. Issuing this command is equivalent to pressing the Reset button. The syntax is:

initialize

After self-tests are executed, the system autoboots unless one of the following is true:

- A halt assertion condition exists (see Section 3.12)

- The **auto_action** environment variable is set to **halt**.

If the **auto_action** environment variable is set to **boot** or **restart** and no halt assertion condition exists, the system autoboots. In all other cases, the system stops in console mode and does not attempt to boot.

4.16 Finding Help

The help (or man) command displays basic information about a command.

Example 4-29 Help (or Man) Command

```
P00>>> help set          # or man set
NAME
      set
FUNCTION
      Set an option or modify the value of an environment
      variable.
SYNOPSIS
      set <option> <value> or <envar> [-] <value>
            where
            <option>={host,mode}
            where
            <envar>={auto_action,bootdef_dev,boot_osflags,...}
            [-default]
```

The **help** (or **man**) command displays basic information about the use of console
commands when the system is in console mode. The syntax is:

help (or **man**) [*command* **. . .**]

command . . .   Command or topic for which help is requested. The options are:

none	Displays the complete list of commands for which you can receive help.
command_name	Displays information about the console command.
argument_string (such as "sh")	Displays information about all commands that begin with that string.

4.17 Switching from SRM to AlphaBIOS Console

It is necessary to switch to the AlphaBIOS console to run configuration utilities. To switch from SRM to AlphaBIOS, issue the alphabios command.

Example 4-30 Switching to the AlphaBIOS Console

```
P00>>> alphabios
```

The **alphabios** command loads and starts the AlphaBIOS console. This is necessary for running AlphaBIOS-based utilities (such as the RAID configuration utility). The syntax is:

alphabios

To return to the SRM console, do one of the following:

- If the **os_type** environment variable is set to **unix** or **openvms**, reset the system by pressing the Reset button.

- If the **os_type** environment variable is set to **nt**, press the Reset button, wait 5 seconds, then press the Halt button. Alternatively, select **DIGITAL UNIX console (SRM)** or **OpenVMS console (SRM)** from the **Advanced CMOS Setup** screen and reset the system.

To switch permanently to the AlphaBIOS console (that is, if the permanent operating system is changed to Windows NT), do the following:

1. Set the SRM **os_type** environment variable to **nt**.

2. Set the SRM **console** environment variable to **graphics**.

3. Reset the system with the Reset button or by power cycling.

4.18 Environment Variable Summary

Environment variables pass configuration information between the console and the operating system. Their settings determine how the system powers up, boots the operating system, and operates. Environment variables are set or changed with the set *envar* command and returned to their default values with the clear *envar* command. Their values are viewed with the show *envar* command.

Table 4-6 lists the environment variables. Detailed descriptions follow. The environment variables are specific to the SRM console.

Table 4-6   Environment Variable Summary

Environment Variable	Function
auto_action	Specifies the console's action at power-up, a failure, or a reset.
boot_osflags	Specifies the default operating system boot flags.
bootdef_dev	Specifies the default boot device string.
com*_baud	Changes the default baud rate of the COM 1 or the COM2 serial port.
console	Specifies the device on which power-up output is displayed (serial terminal or graphics monitor).
cpu_enabled	Enables or disables a specific secondary CPU.
ew*0_mode	Specifies the connection type of the default Ethernet controller.
ew*0_protocols	Specifies network protocols for booting over the Ethernet controller.
kbd_hardware_ type	Specifies the default console keyboard type.
kzpsa*_host_id	Specifies the default value for the KZPSA host SCSI bus node ID.
language	Specifies the console keyboard layout.

Table 4-6   Environment Variable Summary (Continued)

Environment Variable	Function
memory_test	Specifies the extent to which memory will be tested. For DIGITAL UNIX systems only.
ocp_text	Overrides the default OCP display text with specified text.
pci_parity	Disables or enables parity checking on the PCI bus.
pk*0_fast	Enables fast SCSI mode.
pk*0_host_id	Specifies the default value for a controller host bus node ID.
pk*0_soft_term	Enables or disables SCSI terminators on systems that use the QLogic ISP1020 SCSI controller.
tt_allow_login	Enables or disables login to the SRM console firmware on other console ports.

auto_action

Specifies the action the console takes any time the system powers up, fails, or resets. When the setting involves autoboot, the system boots from the default boot device specified by the value of the bootdef_dev environment variable. The syntax is:

set auto_action *value*

The options for *value* are:

halt	The system remains in console mode after power-up or a system crash.
boot	The system boots automatically when it is turned on and halts after a system failure.
restart	The system boots automatically when it is turned on or after it fails.

NOTE: If a halt assertion exists, the console ignores the auto_action setting and halts at the SRM console. See Section 3.12 for information on halt assertion.

bootdef_dev

The **bootdef_dev** environment variable specifies one or more devices for booting the operating system. When more than one device is listed, the system searches in the order listed and boots from the first device with operating system software. The syntax is:

set bootdef_dev *boot_device*

boot_device   The name of the device on which the system software has been loaded. To specify more than one device, separate the names with commas. Enter the command **show bootdef_dev** to display the current default boot device. Enter the command **show device** for a list of all devices in the system.

boot_osflags

The **boot_osflags** environment variable passes information to the **boot** command. That information is dependent on the operating system to be booted.

When the operating system is DIGITAL UNIX, the syntax is:

set boot_osflags *flags_value*

The options for *flags_value* are:

a   Load operating system software from the specified boot device (autoboot). Boot to multi-user mode.

i   Prompt for the name of a file to load and other options (boot interactively). Boot to single-user mode.

s   Stop in single-user mode. Boots /vmunix to single-user mode and stops at the # (root) prompt.

D   Full dump; implies "**s**" as well. By default, if DIGITAL UNIX crashes, it completes a partial memory dump. Specifying "**D**" forces a full dump if the system crashes.

When the operating system is OpenVMS, the syntax is:

set boot_osflags *root_number,boot_flags*

root_number   Directory number of the system disk on which OpenVMS files are located. For example:

root_number	Root Directory
0 (default)	[SYS0.SYSEXE]
1	[SYS1.SYSEXE]
2	[SYS2.SYSEXE]
3	[SYS3.SYSEXE]

boot_flags   The hexadecimal value of the bit number or numbers set. To specify multiple boot flags, add the flag values (logical OR). See Table 4-7.

Table 4-7 Settings for boot_osflags Bootflags (OpenVMS)

Flags_Value	Bit Number	Meaning
1	0	Bootstrap conversationally (enables you to modify SYSGEN parameters in SYSBOOT).
2	1	Map XDELTA to running system.
4	2	Stop at initial system breakpoint.
8	3	Perform diagnostic bootstrap.
10	4	Stop at the bootstrap breakpoints.
20	5	Omit header from secondary bootstrap image.
80	7	Prompt for the name of the secondary bootstrap file.
100	8	Halt before secondary bootstrap.
10000	16	Display debug messages during booting.
20000	17	Display user messages during booting.

com*_baud

The default baud rate for the system is 9600. With the **com*_baud** environment variable, you can set the baud rate to match that of the device connected to the port.

NOTE: You must disable RCM to select a baud rate other than 9600. See Chapter 6.

You will be asked to confirm the change, as shown here:

```
P00>>> set com1_baud 19200
Embedded Remote Console only supports 9600 baud. Continue?
(Y/[N]) n
bad value - com1_baud not modified
P00>>>
```

The syntax is:

set com*_baud *baud_value*

baud_value   The new baud rate. A list of possible values is displayed by attempting to set this environment variable to an unacceptable value (for example, **set com2_baud xxx**).

console

The console terminal can be either a graphics monitor or a serial terminal. The **console** environment variable specifies which is used. The syntax is:

set console *output_device*

The options for *output_device* are:

graphics (default) The console terminal is a graphics monitor or a device connected to the VGA or TGA module.

serial The console terminal is the device connected to the COM1 port.

Whenever you change the value of **console**, you must reset the system by pressing the Reset button or issuing the **initialize** command.

cpu_enabled

Sets a bit mask that enables or disables specific CPUs on a multiprocessor system.

Disabled CPUs are prevented from running the console or the operating system. Bit 0 of the mask corresponds to CPU 0, and bit 1 to CPU 1. A zero in the bit mask prevents the corresponding CPU from running; a one allows it to run. The bit mask is expressed as a hexadecimal value.

NOTE: The primary CPU cannot be disabled.

The syntax is:

set cpu_enabled *hex_digit*

The options for *hex_digit* are 0 – 3 (hexadecimal).

ew*0_mode

Sets an Ethernet controller to run an AUI, ThinWire, or twisted-pair Ethernet network. The default is **auto-sense**. For the fast setting, the device defaults to fast.

The syntax is:

set ew*0_mode *value*

The options for *value* are:

aui	Device type is AUI.
auto-sense	Device type is sensed by the console.
twisted-pair	Device type is 10BaseT (twisted pair).
auto-negotiate	DE500-BA

ew*0_protocols

Enables network protocols for booting and other functions. The syntax is:

set ew*0_protocols *protocol_value*

The options for *protocol_value* are:

mop (default)	Sets the network protocol to mop (Maintenance Operations Protocol), the setting typically used with the OpenVMS operating system.
bootp	Sets the network protocol to bootp, the setting typically used with the DIGITAL UNIX operating system.
bootp,mop	When both are listed, the system attempts to use the mop protocol first, regardless of which is listed first. If not successful, it then attempts the bootp protocol.

kbd_hardware_type

Used only on systems with the language variant 3C (Français), this environment variable sets the keyboard hardware type as either PCXAL or LK411 and enables the system to interpret the terminal keyboard layout correctly.

Whenever you change the value of **kbd_hardware_type**, you must reset the system by pressing the Reset button or issuing the **initialize** command.

The syntax is:

set kbd_hardware_type *keyboard_type*

The options for *keyboard_type* are:

pcxal (default)   Selects the default keyboard hardware type.

lk411   Selects the LK411 keyboard layout for use with language variant
                     3C (Français).

kzpsa*_host_id

See **pk*0_host_id.**

language

Specifies the keyboard layout, which is language dependent. The setting of the **language** environment variable must match the language of the keyboard variant.

Whenever you change the value of **language**, you must reset the system by pressing the Reset button or issuing the **initialize** command.

The syntax is:

set language *language_code*

The options for *language_code* are:

0	No language (cryptic)	**40**	Français (Suisse Romande)
30	Dansk (Danish)	**42**	Italiano (Italian)
32	Deutsch (German)	**44**	Nederlands (Netherlands)
34	Deutsch (Schweiz) (Swiss)	**46**	Norsk (Norwegian)
36	English (American)	**48**	Portuguese (Portuguese)
38	English (British/Irish)	**4A**	Suomi (Finnish)
3A	Español (Spanish)	**4C**	Svenska (Swedish)
3C	Français (French)	**4E**	Belgisch-Nederlands (Dutch)
3E	Français (Canadian)		

memory_test

Determines the extent of memory testing on the next reset. The syntax is:

set memory_test *value*

The options for *value* are:

full (default)   Specifies that the full memory test will be run. Systems using the OpenVMS operating system must run the full memory test.

partial   Specifies that the first 256 MB of memory will be tested.

none   Specifies that memory will not be tested. (However, tests are always run on the first 32 MB.)

ocp_text

Specifies a message to display on the control panel after self-tests and diagnostics have completed.

The value of **ocp_text** takes effect only after you reset the system by pressing the Reset button or issuing the **initialize** command.

The syntax is:

set ocp_text *message*

The variable *message* can be up to 16 characters and must be enclosed in quotation marks.

os_type

The **os_type** environment variable specifies the default operating system. This variable is set at the factory to the setting for the operating system purchased. Use this command to change the factory default setting.

Whenever you change the value of **os_type**, you must reset the system by pressing the Reset button or issuing the **initialize** command.

The syntax is:

set os_type *os_type*

The options for *os_type* are:

unix   DIGITAL UNIX is the default operating system, and the SRM firmware is started during power-up or reset.

openvms   OpenVMS is the default operating system, and the SRM firmware is started during power-up or reset.

nt   Windows NT is the default operating system. The SRM firmware is started, and it loads and starts the AlphaBIOS firmware during power-up or reset. (If a halt assertion condition exists, this environment variable is overridden, and the SRM firmware does not load and start the AlphaBIOS console.)

password

Sets or clears the console password stored in NVRAM.

The syntax is:

set password

The password is not an argument to the **set password** command; the console prompts the user for the string, which must be between 15 and 30 characters.

pci_parity

Disables or enables parity checking on the PCI bus.

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. A side effect of this aberrant behavior is that superfluous PCI parity errors are reported by the host PCI bridge. In such cases, the device can be used as long as parity is not checked; disabling PCI parity checking prevents false parity errors that can cause system problems.

The syntax is:

set pci_parity *value*

The options for *value* are:

on (default)   Enables PCI parity checking.

off   Disables PCI parity checking.

pk*0_fast

Enables fast SCSI to perform in either standard or fast mode. If the system has at least one fast SCSI device, set the default controller speed to fast SCSI (1). Devices on a controller that connects to both standard and fast SCSI devices will perform at the appropriate rate for the device. If the system has no fast SCSI devices, set the default controller speed to standard SCSI (0). If a fast SCSI device is on a controller set to standard, it will perform in standard mode.

The syntax is:

set pk*0_fast *scsi_speed*

The options for *scsi_speed* are:

0   The controller is in standard SCSI mode.

1 (default)   The controller is in fast SCSI mode.

pk*0_host_id

Sets the controller host bus node ID to a value between 0 and 7.

Each SCSI bus in the system requires a controller. Buses can theoretically support up to eight devices; however, the eighth device must always be a controller. Each device on the bus, including the controller, must have a unique ID, which is a number between 0 and 7. This is the bus node ID number.

On each bus, the default bus node ID for the controller is set to 7. You do not need to change the controller bus node ID unless you place two or more controllers on the same bus.

To list the controllers on your system, enter the command **show device**. SCSI devices begin with the letters "pk" (for example, pka0). The third letter is the adapter ID for the controller. When entering the command **set pk*0_host_id**, replace the asterisk with the adapter ID letter.

The syntax is:

set pk*_host_id *scsi_node_id*

The value for scsi_node_id is the bus node ID, a number from 0 to 7.

pk*0_soft_term

Enables or disables SCSI terminators. This command applies to systems that use the QLogic ISP1020 SCSI controller.

The QLogic ISP1020 SCSI controller implements the 16-bit wide SCSI bus. The QLogic module has two terminators, one for the low eight bits and one for the high eight bits.

The syntax is:

set pk*0_soft_term *value*

The options for *value* are:

off	Disables termination of all 16 bits.
low (default)	Enables low eight bits and disables high eight bits.
high	Enables high eight bits and disables low eight bits.
on	Enables all 16 bits.
diff	Places the bus in differential mode.

tt_allow_login

Enables or disables login to the SRM console firmware on alternate console ports. If the environment variable **console** is set to serial, the primary console device is the terminal connected through the COM1 port. The command **set tt_allow_login 1** enables logins through either the COM2 port or a graphics monitor.

The syntax is:

set tt_allow_login *value*

The options for *value* are:

0   Disables login through the COM2 port or a graphics monitor.

1 (default)   Enables login through the COM2 port or a graphics monitor.

Chapter 5

AlphaBIOS Console

AlphaBIOS is the graphical interface that supports the Microsoft Windows NT operating system and some utility programs. This chapter explains how to perform common system management tasks with AlphaBIOS.

Sections in this chapter are:

- Starting AlphaBIOS

- Keyboard Conventions and Help

- Displaying the System Configuration

- Updating Firmware

- Setting Up the Hard Disk

- Performing Setup Tasks

- Installing Windows NT

- Selecting the Version of Windows NT

- Switching from AlphaBIOS to SRM Console

- Running Utility Programs

*NOTE: To run firmware tests on a Windows NT system, switch to the SRM console and issue the **test** command. See Section 5.9.*

5.1 Starting AlphaBIOS

Start AlphaBIOS Setup by pressing F2 from the Boot screen displayed at power-up or reset.

Figure 5-1   Boot Screen

```
                  AlphaBIOS 5.32
Please select the operating system to start:


   Windows NT Server 4.0


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




   digital  AlphaServer1200
                       Family

                    Press <F2> to enter SETUP
```
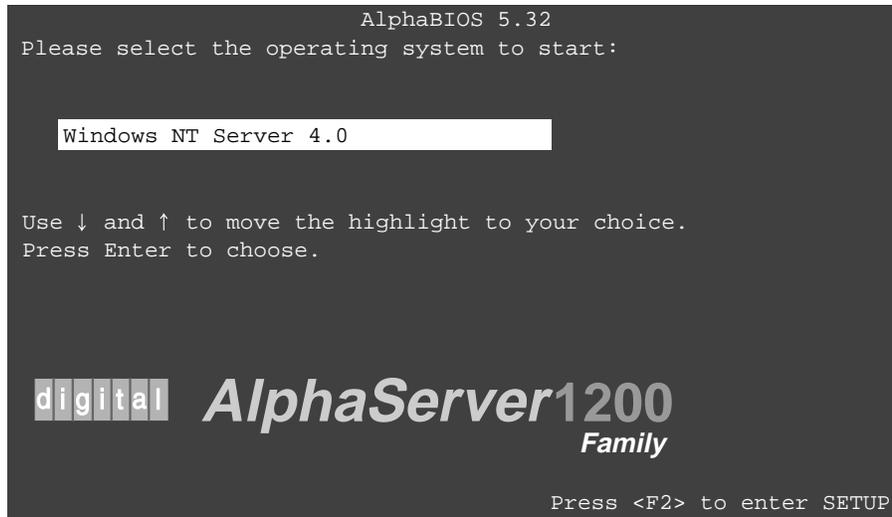
PK-0952A-97

The **Boot** screen shown in Figure 5-1 is displayed at power-up and reset. Press F2 at this screen to enter the setup program.

The **AlphaBIOS Setup** screen (Figure 5-2) is displayed. From this screen you can select the tasks to perform. Use the arrow keys to select the menu item you want and press Enter. (Refer to Section 5.2 for information on navigating the AlphaBIOS screens.)

Figure 5-2   AlphaBIOS Setup Screen

```
                                        AlphaBIOS Setup


      Display System Configuration...
      Upgrade AlphaBIOS
      Hard Disk Setup...
      CMOS Setup...
      Install Windows NT
      Utilities                          ►
      About AlphaBIOS...



    Press ENTER to partition or format hard disks.



  ESC=Exit
```
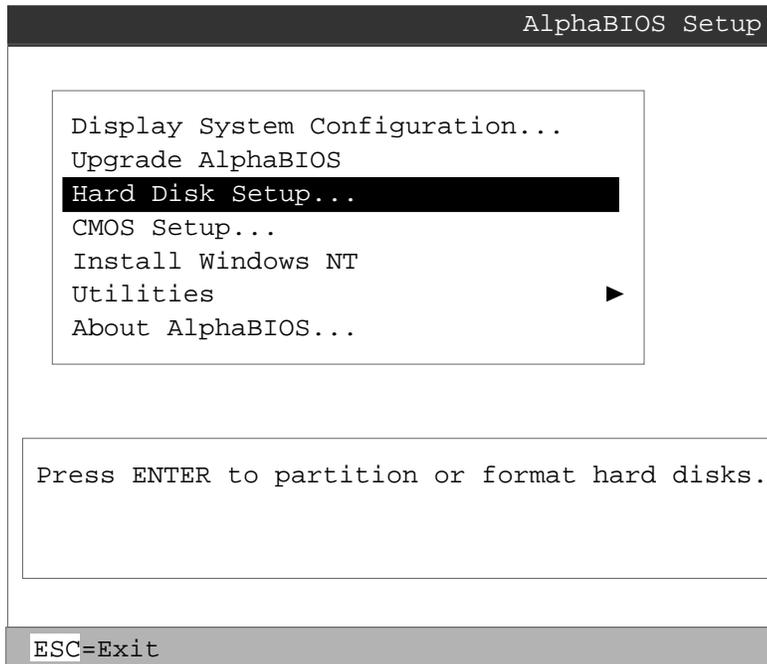
5.2 Keyboard Conventions and Help

AlphaBIOS uses universally accepted keys and key combinations for navigating the interface and selecting items.

Figure 5-3 Typical First-Level Help Screen

```
                  Help: CMOS Setup              F1=Key Help

F3           Change color scheme.


F6           Enter Advanced CMOS Setup.


F7           Set factory default CMOS settings.


ESC          Exit CMOS Setup and discard any changes.


F10          Exit CMOS Setup and save changes,
             including changes from Advanced CMOS Setup.


 ENTER =Continue
```
PK-0724-96

AlphaBIOS uses DOS and Windows keyboard conventions for navigating the interface and selecting items. The valid keystrokes are listed in the keyboard help screens.

Two levels of keyboard help are available. The first level, reached by pressing F1 once, shows explanations of the keystrokes available for the specific part of AlphaBIOS currently displayed. An example of the help displayed by pressing F1 once, in this case from the **CMOS Setup** screen, is shown in Figure 5-3.

The second level of keyboard help, reached by pressing F1 from the first help screen, shows explanations of the keystrokes available for navigating the interface throughout AlphaBIOS (see Figure 5-4).

Figure 5-4   Second-Level Help Screen

```
┌──────────────────────────────────────────────────────────────────────┐
│               AlphaBIOS Setup                              F1=Help     │
├──────────────────────────────────────────────────────────────────────┤
│                          Help: Action Keys                             │
│                                                                        │
│  TAB        Move highlight forward between fields of a dialog.         │
│                                                                        │
│  SHIFT+TAB  Move highlight backward between fields of a dialog.        │
│                                                                        │
│  ↑ ↓        Move highlight within a menu, or cycle through available field │
│             values in a dialog window.                                 │
│                                                                        │
│  ALT+↓      Drop down a menu of choices from a drop-down listbox.  A   │
│             drop-down listbox can be recognized by the ⬇ symbol.       │
│                                                                        │
│  HOME       Move to the beginning of a text entry field.               │
│                                                                        │
│  END        Move to the end of a test entry field.                     │
│                                                                        │
│  ← →        Move to the left or right in a text entry field.           │
│                                                                        │
│  ESC        Discard changes and/or backup to previous screen.          │
│                                                                        │
│                                                                        │
│                                                                        │
├──────────────────────────────────────────────────────────────────────┤
│  ENTER=Continue                                                        │
│                                                                        │
├──────────────────────────────────────────────────────────────────────┤
│                                                                        │
└──────────────────────────────────────────────────────────────────────┘
```

PK-0725-96

5.3 Displaying the System Configuration

The Display System Configuration screen gives information about the system's installed processor, memory, attached devices, and option boards. From the AlphaBIOS Setup screen select Display System Configuration, then the category for the information you need.

Figure 5-5   Display System Configuration Screen

```
                     Display System Configuration


     Systemboard Configuration
     Hard Disk Configuration
     PCI Configuration
     EISA Configuration
     SCSI Configuration
     MC Bus Configuration
     Memory Configuration
     Integrated Peripherals


        System Type:  AlphaServer 1200 5/533
     Processor:  Digital Alpha 21164, Revision 7.1 (2 Processors)
         Speed:   533 MHz
         Cache:     4 MB
        Memory:   512 MB

     Floppy Drive A:  3.5" 1.44 MB
     Floppy Drive B:  None
           Keyboard:  U.S. 101-key keyboard

     AlphaBIOS Version:  5.32  970226.1006
           SRM Version:  V5.0-1, 11-NOV-1997 16:52:39



  ESC=Exit
```

PK-0953-97

Display the system configuration as follows:

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

2. In the **Display System Configuration** screen, use the arrow keys to select the configuration category you want to see.

From this screen, you can view configuration information about these system components:

- System motherboard
- Hard disk
- PCI bus
- EISA bus
- SCSI devices
- System bus
- Memory
- Integrated peripherals

The sections that follow explain the display for each component.

5.3.1 System Board Configuration

Figure 5-6   System Board Configuration

```
                     Display System Configuration           F1=Help

      ┌─────────────────────────────────────┐
      │ Systemboard Configuration          █ │
      │ Hard Disk Configuration            ▓ │
      │ PCI Configuration                  ▓ │
      │ EISA Configuration                 ▓ │
      │ SCSI Configuration                 ▓ │
      │ MC Bus Configuration               ▓ │
      │ Memory Configuration             ▽ │
      └─────────────────────────────────────┘

   ┌──────────────────────────────────────────────────────────┐
   │ ❶    System Type:  AlphaServer 1200 5/533                 │
   │                                                            │
   │ ❷ Processor:  Digital Alpha 21164, Revision 7.1  (2 Processors) │
   │ ❸    Speed:  533 MHz                                       │
   │ ❹    Cache:   4 MB                                         │
   │ ❺   Memory:  512 MB                                        │
   │   Floppy Drive A:  3.5" 1.44 MB                            │
   │   Floppy Drive B:  None                                    │
   │        Keyboard:  U.S. 101-key keyboard                   │
   │                                                            │
   │ ❻ AlphaBIOS Version:  5.32  970226.1006                   │
   │       SRM Version:  V5.0-1, 11-NOV-1997 16:52:39          │
   └──────────────────────────────────────────────────────────┘

 ESC=Exit
```

PK-0954-97

❶ System type — The model number of the system.

❷ Processor — The model and revision of the processor chip. Revision-level information can be useful in troubleshooting problems with technical support personnel.

❸ Speed — The speed of the processor.

❹ Cache — The amount, in kilobytes, of static RAM cache memory installed.

❺ Memory — The amount, in megabytes, of main memory in the system.

❻ Firmware versions — The versions of AlphaBIOS and SRM currently running on the system.

AlphaBIOS Console   **5-9**

5.3.2 Hard Disk Configuration

Figure 5-7 Hard Disk Configuration

```
                    Display System Configuration

    Systemboard Configuration
    Hard Disk Configuration
    PCI Configuration
    EISA Configuration
    SCSI Configuration
    MC Bus Configuration
    Memory Configuration
    Integrated Peripherals


   ❶         ❷       ❸       ❹                  ❺

  Disk  0    NCRC810 #0, SCSI ID 3              510 MB


             Partition 1  ❻                 ❼ 200 MB     ❽ NTFS
             Partition 2                       200 MB       FAT
             Unused                            110 MB

  Disk  1    NCRC810 #0, SCSI ID 5            1001 MB
             DEC    RZ26    (C) DECT386
             Partition 1                        995 MB        NTFS
             Partition 2                          6 MB        FAT
```

PK-0739A-97

NOTE: *This screen is for information only; it cannot be edited. To make changes to the hard disk setup, use the* **Hard Disk Setup** *screen.*

❶ Physical disk ID — Based on the SCSI ID. The disk with the lowest SCSI ID is disk 0, the disk with the next lowest SCSI ID is disk 1, and so on.

❷ Controller — The brand and model of SCSI chip used on the SCSI controller.

❸ Controller number — Based on how many SCSI controllers of a particular type are installed in the system. The first controller of a type is always numbered 0.

❹ SCSI ID number — A unique number you assign to each SCSI device installed in the system. This is usually done with jumpers or a thumb wheel attached to the drive housing.

❺ Size — The raw capacity of the drive. Formatting the drive with different file systems (for example, FAT and NTFS) may result in different usable sizes because of the differences in how storage is managed under those file systems.

❻ Partition number — Within a single drive, partition numbers are assigned in sequential order: 1, 2, 3, and so on. The partitions populate the drive from the innermost cylinders to the outermost cylinders. If you have a large hard disk (over 800 MB) and plan to use the FAT file system, it is a good idea to break the disk into several smaller partitions because the FAT file system uses disk space more efficiently at smaller partition sizes. This is not a concern for the NTFS file system, however, as it uses disk space very efficiently at all partition sizes.

❼ Partition size — The raw (unformatted) storage capacity of the partition. Actual storage space will differ based on the file system with which the partition is formatted.

❽ Partition format — The file system (if any) used on a partition. This field displays FAT, NTFS, or unrecognized (if the partition is unformatted).

5.3.3 PCI Configuration

Figure 5-8 PCI Configuration

```
┌──────────────────────────────────────────────────────────────────┐
│        Display System Configuration                    F1=Help     │
├────────────────────────────────────────────────────────────────────┤
│                                                                     │
│    ┌─────────────────────────────────────┐▲                        │
│    │ Systemboard Configuration           │█                        │
│    │ Hard Disk Configuration             │█                        │
│    │ PCI Configuration                   │█                        │
│    │ EISA Configuration                  │                         │
│    │ SCSI Configuration                  │                         │
│    │ MC Bus Configuration                │                         │
│    │ Memory Configuration              ▼ │                         │
│    └─────────────────────────────────────┘                        │
│         ❶           ❷              ❸             ❹                  │
│  ┌─────────────────────────────────────────────────────────────┐  │
│  │ Device Name      Device Type    Revision   Physical Slot      │  │
│  │ ───────────      ───────────    ────────   ─────────────      │  │
│  │                                                             █ │  │
│  │ Intel 82375 PCEB EISA bridge    5          PCI0-Embedded    █ │  │
│  │ S3 Trio32/64     VGA            0          PCI0-2             │  │
│  │ DIGITAL 21140    Ethernet       32         PCI0-3             │  │
│  │ NCRC810          SCSI           2          PCI1-Embedded      │  │
│  │ DIGITAL KZPSX    SCSI           0          PCI1-2             │  │
│  │ DIGITAL KZPSX    SCSI           0          PCI1-3             │  │
│  │ DIGITAL 21050    PCI bridge     2          PCI1-4             │  │
│  │ QLogic ISP1020/1040 SCSI        2          PCI1-4-Bridged  ▼  │  │
│  └─────────────────────────────────────────────────────────────┘  │
│  ┌──────────────────────────────┐                                  │
│  │ ENTER=Select   ESC=Exit       │                                 │
│  └──────────────────────────────┘                                  │
└────────────────────────────────────────────────────────────────────┘
```

PK-0740D-97

❶ Device name — The name and model of the device as recorded in the device's firmware.

❷ Device type — Lists the function of the device in the system.

❸ Revision — The revision level of the device signifies the number of times it has been updated by the manufacturer.

❹ Physical slot — Lists the actual PCI slot number to which the device is attached.

You can find additional detail about any of the PCI devices listed in the **Advanced PCI Information** screen. Follow this procedure to view this screen:

1. Press Enter to enable selection in the device list.

2. Use the arrow keys to select the device for which you want additional detail.

3. Press Enter, and the detail is displayed.

4. Press Escape to return to the table of PCI devices.

The **Advanced PCI Information** screen is shown in Figure 5-9 on the next page.

Figure 5-9 Advanced PCI Information

```
                    Advanced PCI Information
❶                         ❷                    ❸
Bus Number=0, Device Number=7, Function Number=0
❹
Configuration Space:

    Register Name              Hex Offset   Hex    Value

    Vendor ID                      00              8086
    Device ID                      02              0482
    Command                        04              0047
    Status                         06              0200
    Revision ID                    08                05
    Prog. I/F                      09                00
    Sub Class Code                 0A                00
    Class Code                     0B                00
    Cache Line Size                0C                00
    Latency Timer                  0D                20
    Header Type                    0E                00
    BIST                           0F                00
    Base Address 0                 10          00000000   ▼


ENTER=Continue
```

PK-0741A-96

❶ Bus number — The virtual PCI bus number.

❷ Device number — The PCI bus device number.

❸ Function number — Represents the number assigned to a particular function on
   a multifunction device. For example, a combination Ethernet/SCSI controller
   would be listed twice, with the first function listed as 0 and the other as 1.

❹ Configuration header space — Displays the information in the selecte d device's
   PCI configuration space.

5.3.4 EISA Configuration

Figure 5-10 EISA Configuration

```
                        Display System configuration

      ┌─────────────────────────────────────────────┐
      │ Systemboard Configuration                 ▲  │
      │ Hard Disk Configuration                   ░  │
      │ PCI Configuration                         ░  │
      │ EISA Configuration                        ░  │
      │ SCSI Configuration                        ░  │
      │ MC Bus Configuration                      ░  │
      │ Memory Configuration                      ░  │
      │ Integrated Peripherals                    ▼  │
      └─────────────────────────────────────────────┘

          ❶                      ❷                  ❸
      ┌──────────────────────────────────────────────────┐
      │ Device Name           Device Type      Physical Slot │
      │ ─────────────         ───────────      ───────────── │
      │                                                      │
      │ DEC 5301              Other            Embedded      │
      │ FLOPPY                DISK             Embedded      │
      │                                                      │
      └──────────────────────────────────────────────────┘

       ENTER=Select   ESC=Exit
```
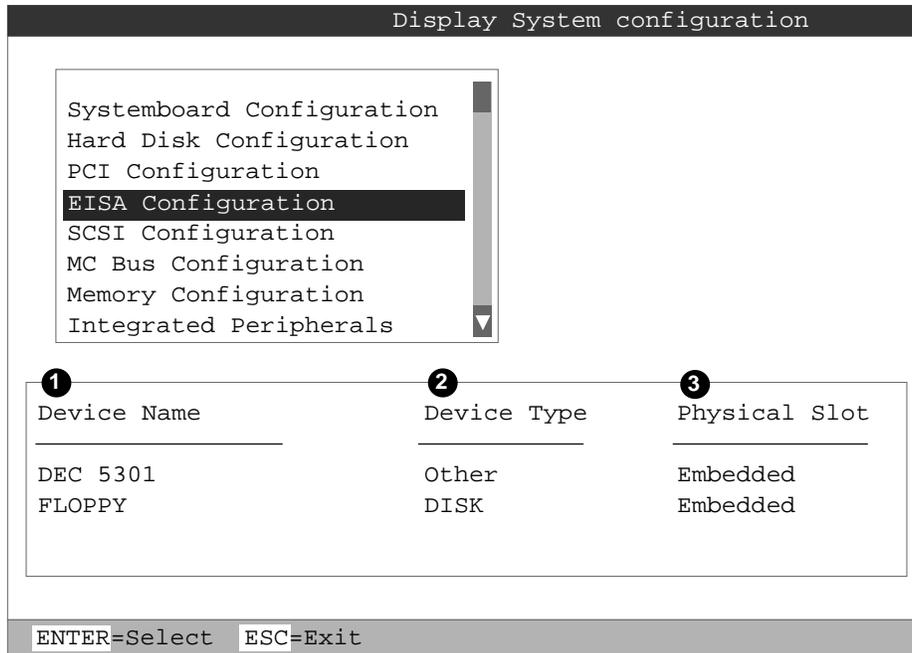
PK-0742A-97

❶ Device name — Includes a three-character manufacturer code, followed by a three-digit board type, followed by a one-digit revision number.

❷ Device type — Identifies the board type (for example, network).

❸ Physical slot — Lists the actual EISA slot number to which the de vice is attached. The embedded EISA system controller is always first on this list.

5.3.5 SCSI Configuration

Figure 5-11   SCSI Configuration

```
┌──────────────────────────────────────────────────────────────────┐
│                   Display System Configuration                     │
│                                                                    │
│  ┌─────────────────────────────────┐                               │
│  │ Systemboard Configuration       │                               │
│  │ Hard Disk Configuration         │                               │
│  │ PCI Configuration               │                               │
│  │ Eisa Configuration              │  ┌────────────────┐           │
│  │ SCSI Configuration           ▶  │  │ QLISP1020 #0   │           │
│  │ MC Bus Configuration            │  │ KZPSX #0       │           │
│  │ Memory Configuration            │  └────────────────┘           │
│  │ Integrated Peripherals          │                               │
│  └─────────────────────────────────┘                               │
│                                                                    │
│  ┌──────────────────────────────────────────────────────────────┐ │
│  │ ❶  QLISP1020 #0, SCSI ID 7, SCSI Bus 0                         │ │
│  │ ─────────────────────────────────────────────────────────────│ │
│  │    ❷        ❸              ❹         ❺                          │ │
│  │  SCSI ID  Device          Size      Description                │ │
│  │ ─────────────────────────────────────────────────────────────│ │
│  │    0      Disk 0         2047 MB    SEAGATE   ST32550W  0021   │ │
│  │    1      Disk 1         2049 MB    SEAGATE   ST32551W  0526   │ │
│  │    5      CD-ROM                    DEC       RRD45 (C) DEC1645 │ │
│  └──────────────────────────────────────────────────────────────┘ │
└──────────────────────────────────────────────────────────────────┘
                                                         PK-0743A-96
```

❶ SCSI controller information — Describes the physical characteristics of the selected SCSI controller. This line includes:

Controller — Brand and model of SCSI chip used on the SCSI controller.

Controller number — Based on the number of SCSI controllers of a particular type in the system. The first controller of a type is always numbered 0.

SCSI ID number — A unique number assigned to the SCSI controller. The standard scheme in numbering controllers is for controllers to be SCSI ID 7.

SCSI bus number — Indicates if the controller is first or second in the system. The first controller is tied to SCSI bus 0, and the second to SCSI bus 1.

❷ SCSI ID — A unique number you assign to each SCSI device in the system. This is usually done with jumpers or a thumb wheel attached to the drive housing.

❸ Device type — Displays whether the device is a hard disk, CD-ROM, scanner, or other type.

❹ Size — The raw capacity of the drive. Formatting the drive with different file systems (for example, FAT and NTFS) may result in different usable sizes because of differences in the way those file systems manage storage. This field is left blank if it is not applicable to a device (for example, a scanner).

❺ Description — The name and model of the d evice as recorded in the device's firmware.

5.3.6 System Bus Configuration

Figure 5-12 System Bus Configuration

```
┌──────────────────────────────────────────────────────────────────┐
│            Display System Configuration          F1=Help           │
│                                                                    │
│   ┌──────────────────────────────────────┐▲                       │
│   │ Systemboard Configuration            │█                        │
│   │ Hard Disk Configuration              │                         │
│   │ PCI Configuration                    │                         │
│   │ EISA Configuration                   │                         │
│   │ SCSI Configuration                   │█                        │
│   │ MC Bus Configuration                 │                         │
│   │ Memory Configuration                 │▼                        │
│   └──────────────────────────────────────┘                        │
│                                                                    │
│    ❶     ❷              ❸          ❹            ❺                  │
│  ┌────────────────────────────────────────────────────────────┐   │
│  │ McBus  Module Name        Module Type    Revision   Physical Slot │
│  │ ────   ───────────        ───────────    ────────   ───────────── │
│  │ 1      Memory             0              0000       MEM        │   │
│  │ 2      CPU (EV56 Cached)  3              0002       CPU0       │   │
│  │ 3      CPU (EV56 Cached)  3              0002       CPU1       │   │
│  │ 4      MCBus-PCI Bridge   6008           0031       IOD0/1     │   │
│  │ 5      MCBus-PCI Bridge   6000           0031       IOD0/1     │   │
│  └────────────────────────────────────────────────────────────┘   │
│                                                                    │
│  ESC=Exit                                                          │
└──────────────────────────────────────────────────────────────────┘
```

PK-0744D-97

❶ MC bus number — Identifies the location of a component on the system bus.

❷ Module name — The DIGITAL internal identifier for a system component.

❸ Module type — Identifies the function of the system component.

❹ Revision — A DIGITAL internal number representing the manufacturing revision level of the system component.

❺ Physical slot — Represents the actual location of the system component on the bus.

5.3.7 Memory Configuration

Figure 5-13   Memory Configuration

```
┌──────────────────────────────────────────────────────────────────────────┐
│          Display System Configuration                          F1=Help     │
├──────────────────────────────────────────────────────────────────────────┤
│   ┌────────────────────────────────┐                                       │
│   │ Systemboard Configuration      │                                       │
│   │ Hard Disk Configuration        │                                       │
│   │ PCI Configuration              │                                       │
│   │ EISA Configuration             │                                       │
│   │ SCSI Configuration             │                                       │
│   │ MC Bus Configuration           │                                       │
│   │ Memory Configuration           │                                       │
│   │ Integrated Peripherals         │                                       │
│   └────────────────────────────────┘                                       │
│                                                                            │
│   ┌──────────────────────────────────────────────────────────────────┐    │
│   │                                                                    │    │
│   │                   System Memory Configuration                     │    │
│   │                   ─────────────────────────────                    │    │
│   │                                                                    │    │
│   │                                                                    │    │
│   │    Bank 0: 256 MB (128 MB Per DIMM) -- Starting Address = 0X00000000│   │
│   │    Bank 1:  64 MB ( 32 MB Per DIMM) -- Starting Address = 0X20000000│   │
│   │                                                                    │    │
│   └──────────────────────────────────────────────────────────────────┘    │
│                                                                            │
├──────────────────────────────────────────────────────────────────────────┤
│ ESC=Exit                                                                   │
└──────────────────────────────────────────────────────────────────────────┘
```

PK-0745B-97

5.3.8 Integrated Peripherals

Figure 5-14 Integrated Peripherals

```
                           Display System Configuration

     Systemboard Configuration
     Hard Disk Configuration
     PCI Configuration
     Eisa Configuration
     SCSI Configuration
     MC Bus Configuration
     Memory Configuration
     Integrated Peripherals


        ❶                   ❷                   ❸   ❹
     Serial Port 1: Enabled as COM1:  3F8, IRQ4

     Serial Port 2: Enabled as COM2:  2F8, IRQ3

     Parallel Port 1: Enabled as LPT1:  3BC, IRQ7
```

PK-0746A-96
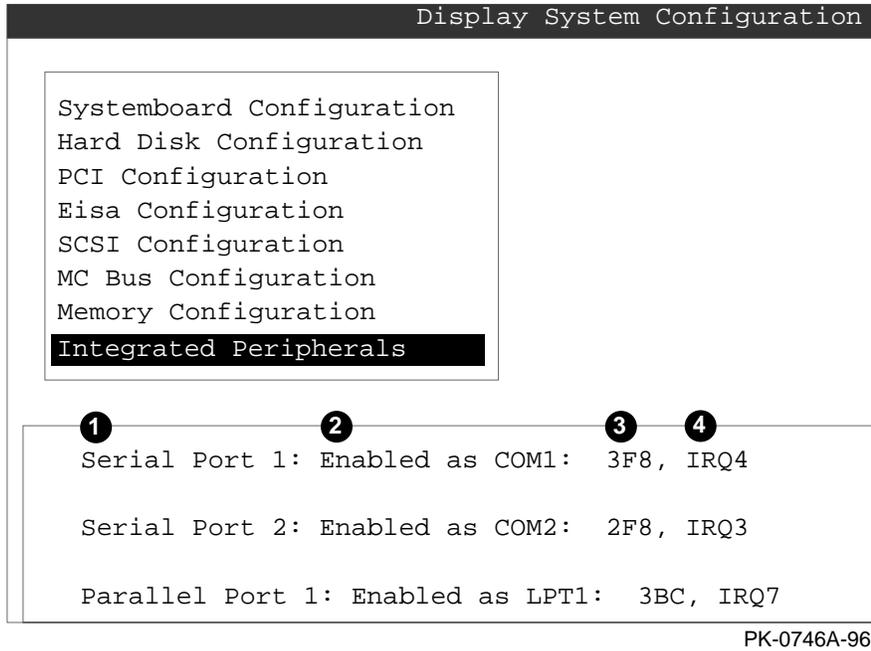
❶ Device type — The physical device as it exists on the system board.

❷ MS DOS name — Shows if the device is enabled, and if it is enabled, the addressable MS-DOS name for the device.

❸ Port address — The physical memory location for data received into the device, and sent from the device, respectively.

❹ Interrupt — The interrupt request line (IRQ) used by the device to get the CPU's attention.

5.4 Updating Firmware

Insert the CD-ROM or diskette with the updated firmware and select Upgrade AlphaBIOS from the main AlphaBIOS Setup screen. Use the Loadable Firmware Update Utility (LFU) to perform the update. The LFU exit command causes a system reset.

Figure 5-15   Updating Firmware

```
                              AlphaBIOS Setup

       ┌─────────────────────────────────────────────┐
       │ Display System Configuration...              │
       │ Upgrade AlphaBIOS                             │
       │ Hard Disk Setup                               │
       │ CMOS Setup...                                 │
       │ Install Windows NT                            │
       │ Utilities                             ►       │
       │ About AlphaBIOS...                            │
       └─────────────────────────────────────────────┘


     ┌───────────────────────────────────────────────────┐
     │ Press ENTER to upgrade your AlphaBIOS from floppy or CD-ROM. │
     │                                                     │
     │                                                     │
     └───────────────────────────────────────────────────┘

       ESC=Exit
```
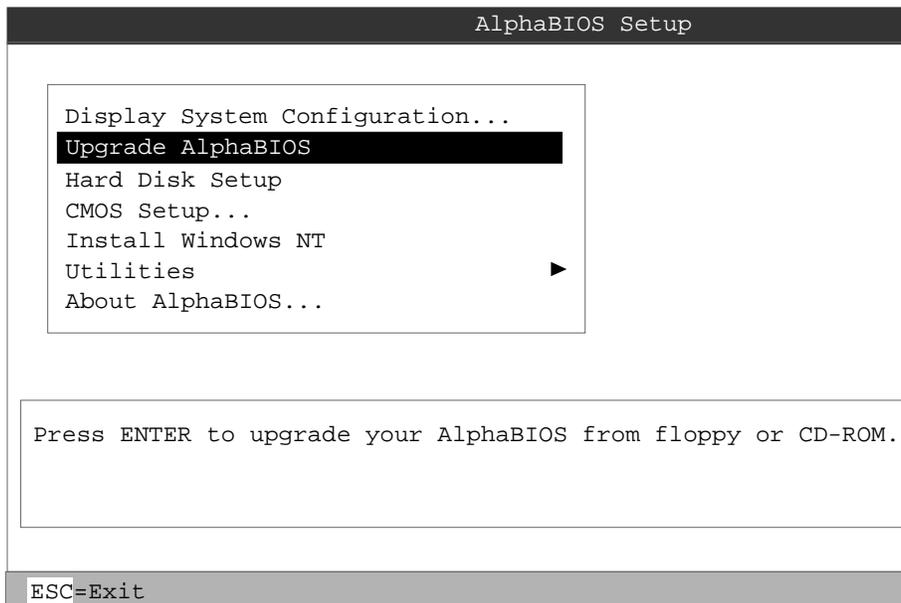
PK-0726A-96

As new versions of Windows NT are released, it might be necessary to upgrade AlphaBIOS to the latest version. Also, as improvements are made to AlphaBIOS, you may wish to upgrade to take advantage of new features.

Use this procedure to upgrade from an earlier version of AlphaBIOS:

1. Insert the diskette or CD-ROM containing the AlphaBIOS upgrade.

2. If you are not already running **AlphaBIOS Setup**, restart your system and press F2 when the **Boot** screen is displayed.

3. In the main **AlphaBIOS Setup** screen, select **Upgrade AlphaBIOS** and press Enter.

    The system is reset and the Loadable Firmware Update (LFU) utility is started.

4. When the upgrade is complete, issue the LFU **exit** command. The system is reset and you are returned to AlphaBIOS.

    If you press the Reset button instead of issuing the LFU **exit** command, the system is reset and you are returned to LFU.

5.5 Setting Up the Hard Disk

You can perform either an express or a custom hard disk setup. An express setup, described in this section, creates the recommended partition arrangement on the first hard disk, but it does not map bad sectors. Custom hard disk setup is described in Sections 5.5.1 and 5.5.2.

Figure 5-16   Hard Disk Setup Screen

```
                              Hard Disk Setup
❶            ❷         ❸   ❹                        ❺
Disk   0     QLISP1020 #0, SCSI ID 3                  510 MB
        ❻ Partition 1                         ❼ 200 MB      ❽ NTFS
          Unused                                310 MB

Disk   1     NCRC810 #0, SCSI ID 5               1001 MB
             Partition 1                          995 MB           NTFS
             Partition 2                            6 MB           FAT
❾
INSERT=New   DEL=Delete   F6=Format   F7=Express   ESC=Exit
```

PK-0732A-96

CAUTION: Hard disk changes are immediate; the changes are made as soon as they are entered. Unintended data loss could occur, so use care when changing your hard disk arrangement.

❶ Physical disk ID — Based on the SCSI ID. The disk with the lowest SCSI ID is disk 0, the disk with the next lowest SCSI ID is disk 1, and so on.

❷ Controller — Brand and model of SCSI chip used on the SCSI controller.

❸ Controller number — Based on how many SCSI controllers of a particular type are installed in the system. The first controller of a type is always numbered 0.

❹ SCSI ID number — A unique number you assign to each SCSI device in the system. This is usually done with jumpers or a thumb wheel attached to the drive housing.

❺ Size — The raw capacity of the drive. Formatting the drive with different file systems (for example, FAT and NTFS) may result in different usable sizes because of the differences in how storage is managed under those file systems.

❻ Partition number — Within a single drive, partition numbers are assigned in sequential order: 1, 2, 3, and so on. The partitions populate the drive from the innermost cylinders to the outermost. If you have a large hard disk (over 800 MB) and plan to use the FAT file system, it is a good idea to break the disk into several smaller partitions because the FAT file system uses disk space more efficiently at smaller partition sizes. This is not necessary for the NTFS file system, since it uses disk space very efficiently at all partition sizes.

❼ Partition size — The raw (unformatted) storage capacity of the partition. Actual storage space will differ based on the file system with which the partition is formatted.

❽ Partition format — The file system (if any) used on a partition. This field displays FAT, NTFS, or unrecognized (if the partition is unformatted).

❾ Disk setup options

Insert partition (Insert key) — Use this option to create new partitions. Before creating a new partition, be sure to select an unpartitioned space. Pressing the Insert key while an already partitioned space is selected causes an informational error to be displayed.

Delete partition (Delete key) — Use this option to delete existing partitions. Before deleting a partition, be sure to back up any data you want to save on the partition. Deleting a partition deletes all data on that partition but leaves the rest of the disk unaffected. Pressing the Delete key while an unpartitioned space is selected displays an informational error.

Format partition (F6 key) — Format a partitioned space with the FAT file system.

Express setup (F7 key) — Create the default recommended partition arrangement for Windows NT.

Exit (Escape key) — Return to the **AlphaBIOS Setup** screen.

An express hard disk setup creates the recommended partition arrangement on the first hard disk (disk 0). It does not, however, format the large partition with NTFS, and it does not map bad sectors on the disk. The NTFS formatting can be done during Windows NT installation.

To perform an express hard disk setup

1. If you have any needed information on your disk, back it up before proceeding.

2. Start **AlphaBIOS Setup**, select **Hard Disk Setup**, and press Enter.

3. Press F7 to enter **Express Setup**.

4. Press F10 to continue with the setup.

5.5.1 Creating and Deleting Partitions Manually

Use the Create Partition and Delete Partition options if you need to create a custom hard disk partition arrangement or otherwise manually manage your hard disk partitions.

Figure 5-17 Create New Partition Dialog Box

```
                        Hard Disk Setup

 Disk  0    QLISP1020 #0, SCSI ID 3               510 MB
            Partition 1                           200 MB          NTFS
            Unused                                310 MB
            Create new partition, Disk 0 Partition 2
 Disk  1
                                                                 NTFS
                                                                 FAT

            Available space:   310   MB

            New Partition Size:  200    MB



            ENTER=Continue   ESC=Cancel
```
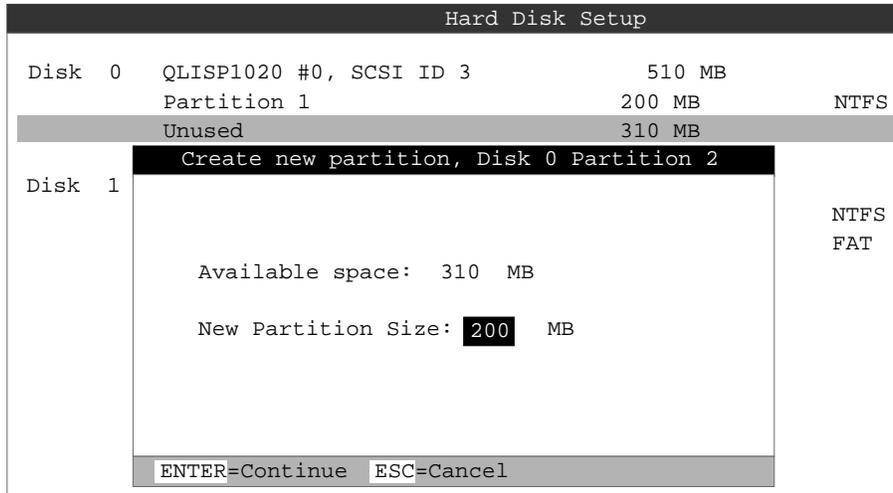
PK-0733-96

To create a partition

1. Start AlphaBIOS and select **Hard Disk Setup**. Press Enter.

2. Select the disk on which to create the partition.

3. If one or more partitions already exist on the disk, select the unpartitioned space.

4. Press Insert. A dialog box is displayed, similar to Figure 5-17.

5. Type the size of the partition to create and press Enter.

To delete a partition

1. Start AlphaBIOS and select **Hard Disk Setup**. Press Enter.

2. Select the partition to be deleted (see ❶ in Figure 5-18).

3. Press Delete. A dialog box is displayed (see ❷).

4. Press F10 to confirm the deletion.

Figure 5-18   Delete Partition Dialog Box

```
                            Hard Disk Setup

 Disk  0   QLISP1020 #0, SCSI ID 3              510 MB
           Partition 1                         200 MB           NTFS
     ❶    Partition 2                         200 MB           FAT
           Unused                              110 MB

 Disk  1   NCRC810 #0, SCSI ID 5             1001 MB
           Partition 1                         995 MB           NTFS
           Partitio┌──────── Delete Disk 0, Partition 2 ────────┐  FAT
                    │                                            │
              ❷    │ The delete operation will destroy all      │
                    │ data on partition 2 of disk 0.             │
                    │                                            │
                    │ Press F10 to continue.  Press ESC to       │
                    │ cancel.                                    │
                    ├────────────────────────────────────────────┤
                    │ F10=Continue   ESC=Cancel                  │
                    └────────────────────────────────────────────┘
```

PK-0734A-96

AlphaBIOS Console   **5-27**
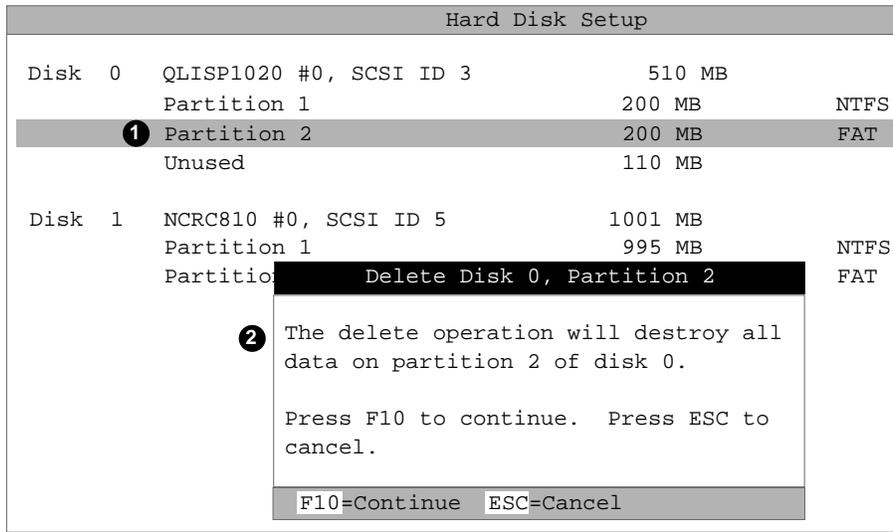
5.5.2 Formatting a FAT Partition

AlphaBIOS can format partitions with the FAT file system. Use Windows NT to format a partition using NTFS.

Figure 5-19   Formatting a FAT Partition

```
                          Hard Disk Setup                        F1=Help

  Disk  0    QLISP1020 #0, SCSI ID 3              510 MB
             Partition 1                          200 MB          NTFS
      ❶      Partition 2                          200 MB          Unrecognized

                   Delete Disk 0, Partition 2
  Disk  1
                                                                 NTFS
      ❷      Choose Format Method:                               FAT

             Format Method: Standard Format ▼

                            Standard Format
                            Quick Format


       ENTER=Continue   ESC=Cancel
```
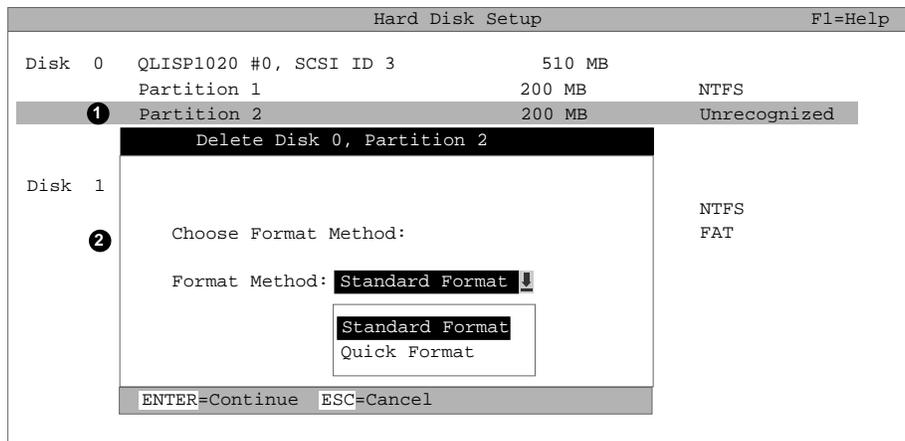
PK-0735A-96

To format a FAT partition

1. Start AlphaBIOS and select **Hard Disk Setup**. Press Enter.

2. Select the partition to be formatted (see ❶ in Figure 5-19).

3. Press F6. A dialog box is displayed, asking whether to perform a quick or standard format (see ❷). If you select **Quick Format**, the formatting is completed immediately, but no bad sectors are mapped. If you select **Standard Format**, a dialog box similar to that in Figure 5-20 is displayed while the drive is formatted, showing the progress of the formatting. Standard formatting maps bad sectors.

4. Select a format method, and press Enter.

Figure 5-20   Standard Formatting

```
                              Hard Disk Setup                        F1=Help

  Disk  0    QLISP1020 #0, SCSI ID 3               510 MB
             Partition 1                           200 MB          NTFS
             Partition 2                           200 MB          Unrecognized
             Unused                                110 MB

  Disk  1    NCRC810 #0, SCSI ID 5               1001 MB
             Partition 1                           995 MB          NTFS
             Partition 2                             6 MB          FAT
                    Disk 0, Partition 2          200 MB

                            Formatting...

                      ┌──────┬─────────────────────┐
                      │      │       25%           │
                      └──────┴─────────────────────┘


                     ESC=Cancel
```
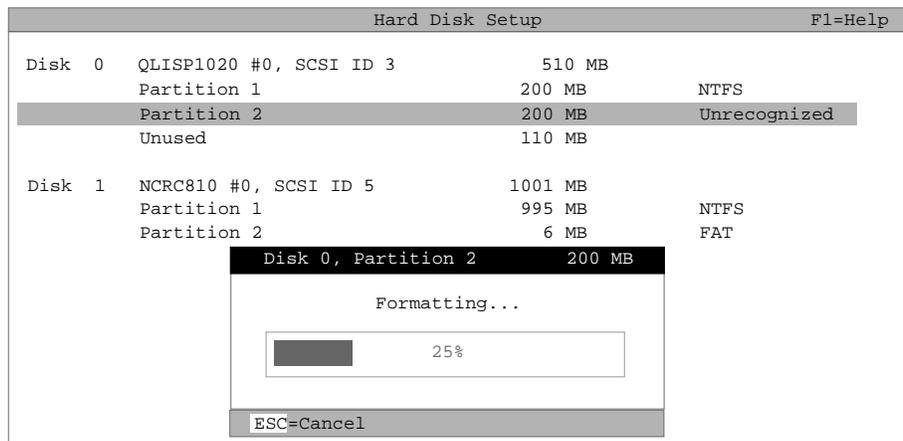
PK-0736-96

5.6 Performing Setup Tasks

CMOS Setup is used to configure several system parameters. CMOS Setup has two modes: Standard CMOS Setup is used to configure basic system parameters; Advanced CMOS Setup is used for system-specific parameters and password protection.

Figure 5-21   Standard CMOS Setup Screen

```
                              CMOS Setup                      F1=Help


 ❶ Date:  Friday,     May       ▼ 10  1997
    Time:  13:22:27



 ❷ Floppy Drive A: 3.5" 1.44 MB
    Floppy Drive B: None
 ❸ Keyboard:  U.S. 101-key keyboard


 ❹ Auto Start: Enabled
    Auto Start Count:  30 Seconds


  ┌─────────────────────────────────────────────────────────────┐
  │ Press ↑ or ↓ to modify date fields.  Date modifications will  │
  │ take effect immediately.                                      │
  │                                                               │
  └─────────────────────────────────────────────────────────────┘
 ❺
 F3=Color  F6=Advanced  F7=Defaults  ESC=Discard Changes  F10=Save Changes
```

PK-0749C-97
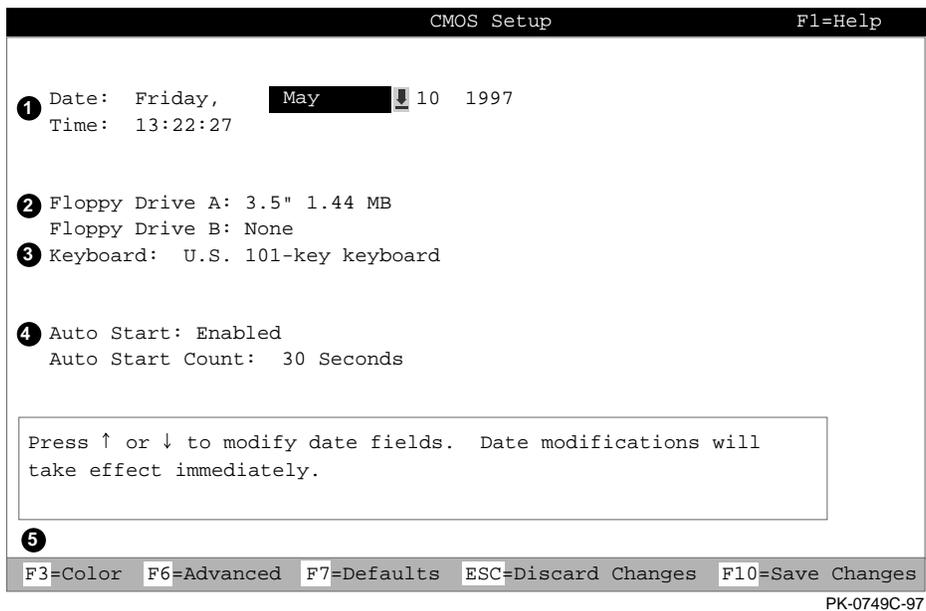
To enter Standard CMOS Setup

Start AlphaBIOS, select **CMOS Setup**, and press Enter.

❶ Date and time — When setting the time, use the 24-hour format. (For example, 10:00 p.m. is 22:00:00.)

❷ Floppy drive — The only drive type supported is 3.5 inch, 1.44 MB.

❸ Keyboard — The keyboard setting makes it possible to use most language keyboards. To ensure correct character mappings, the language of your keyboard, Windows NT, and the keyboard language selection in **CMOS Setup** should all match.

❹ Auto start and auto start count — The **Auto Start** setting determines whether the primary operating system is automatically started after the system is reset or power-cycled. The Auto Start Count setting is the amount of time the **Boot** screen is displayed before the default system is automatically started. This delay gives you the opportunity, after resetting or power-cycling the system, to select another version of Windows NT to start or to enter **AlphaBIOS Setup**.

❺ **Standard CMOS Setup** options

Color (F3 key) — Repeatedly cycles through the available AlphaBIOS color schemes.

Advanced (F6 key) — Displays the **Advanced CMOS Setup** screen.

Defaults (F7 key) — Restores the default **Standard CMOS Setup** values without affecting the **Advanced CMOS Setup** values.

Discard changes (Escape key) — Restores the settings in effect when you started **CMOS Setup**. This option also discards changes made in **Advanced CMOS Setup**.

Save changes (F10 key) — Saves changes made in both the standard and advanced modes of **CMOS Setup**.

Figure 5-22   Advanced CMOS Setup Screen

```
┌─────────────────────────────────────────────────────────────────────────┐
│                      Advanced CMOS Setup                    F1=Help       │
├───────────────────────────────────────────────────────────────────────── │
│                                                                           │
│ ❶ PCI Parity Checking:  Disabled                                          │
│                                                                           │
│ ❷ Power-up Memory Test:  Enabled                                          │
│                                                                           │
│ ❸ AlphaBIOS Password Option:  Disabled                                    │
│                                                                           │
│ ❹ Console Selection:  ┌───────────────────────────────────┐              │
│                       │ Windows NT Console (AlphaBIOS)    │              │
│ ❺ Power-up Memory Test│ Windows NT Console (AlphaBIOS)    │              │
│                       │ OpenVMS Console (SRM)             │              │
│                       │ Digital UNIX Console (SRM)        │              │
│                       └───────────────────────────────────┘              │
│                                                                           │
│                                                                           │
│   ┌─────────────────────────────────────────────────────────────────┐    │
│   │ Press ↑ or ↓ to select the firmware console that will be presented│  │
│   │ the next time the system is power-cycled.                         │    │
│   │                                                                   │    │
│   └─────────────────────────────────────────────────────────────────┘    │
│                                                                           │
│  ❻                                                                        │
│ ESC=Discard Changes   F10=Save Changes                                    │
└───────────────────────────────────────────────────────────────────────── ┘
```

PK-0750A-96

To Enter Advanced CMOS Setup

1. Start AlphaBIOS, select **CMOS Setup**, and press Enter.

2. In the **Standard CMOS Setup** screen, press F6.

❶ PCI parity checking — Enables and disables settings for PCI parity checking, which ensures data integrity across the PCI bus. Because some third-party PCI options do not correctly implement PCI parity generation, the default is Disabled.

❷ Power-up memory test — Enables and disables settings for the power-up memory test.

❸ Password setup — Enables and disables settings for password setup.

Password protection provides two levels of security for your system: setup, which requires a password to start **AlphaBIOS Setup**, and startup, which requires a password before the system initializes. Startup protection provides more comprehensive protection than setup protection because the system cannot be used at all until the correct password is entered.

❹ Console selection — Switches from AlphaBIOS to SRM firmware. If you select OpenVMS (SRM) or DIGITAL UNIX (SRM), the next time you reset your system, the SRM firmware will be loaded instead of AlphaBIOS. This menu selection changes the setting of the **os_type** environment variable in the SRM console.

❺ Memory test length — Selects the length of memory testing that will be done the next time the system is power-cycled.

❻ **Advanced CMOS Setup** options

Discard changes (Escape key) —Restores those settings in effect when you started **Advanced CMOS Setup**. This does not discard changes made to **Standard CMOS Setup**.

Save Changes (F10 key) — Saves changes made in **Advanced CMOS Setup**. When exiting **CMOS Setup**, you must also save your changes at the **Standard CMOS Setup** screen for the changes to be stored permanently.

5.7 Installing Windows NT

Install Windows NT from CD-ROM. Up to four versions of the operating system can be resident in the system at one time.

Figure 5-23   Installing Windows NT

```
                                              AlphaBIOS Setup

        ┌────────────────────────────────────────────────┐
        │                                                │
        │  Display System Configuration...               │
        │  Upgrade AlphaBIOS                              │
        │  Hard Disk Setup...                             │
        │  CMOS Setup...                                  │
        │  Install Windows NT                             │
        │  Utilities                              ▶       │
        │  About AlphaBIOS...                             │
        │                                                │
        └────────────────────────────────────────────────┘


        ┌────────────────────────────────────────────────┐
        │  Press ENTER to install Windows NT.            │
        │                                                │
        │                                                │
        └────────────────────────────────────────────────┘

         ESC=Exit
```
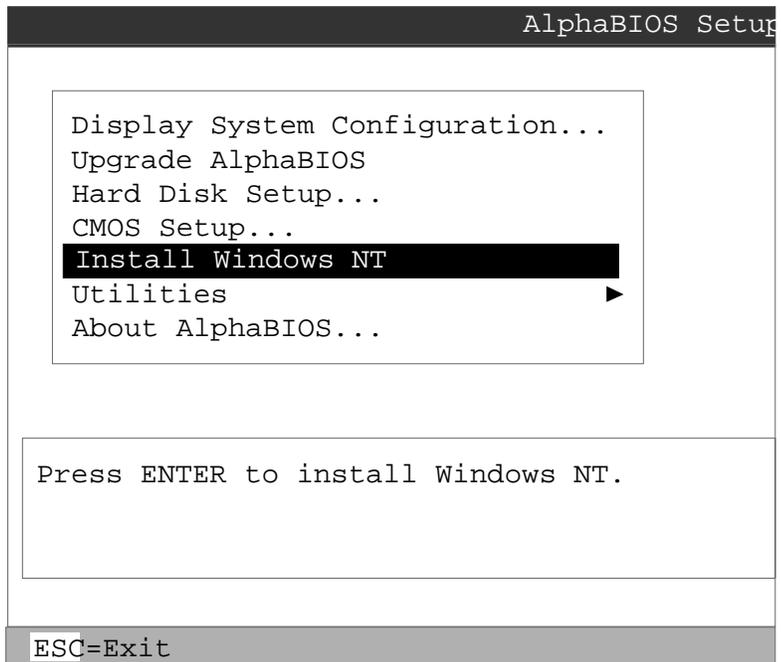
PK-0726B-96

If Windows NT was installed at the factory, Windows NT setup will start automatically the first time the system powers up.

NOTE: Steps 1 and 2 in the following procedure are necessary only when you are first setting up your system. On subsequent installations and upgrades, begin at step 3.

1.   Use **CMOS Setup** to set the system date and time: start **AlphaBIOS Setup**, select **CMOS Setup**, and press Enter.

2.   Perform an express hard disk setup: return to the main **AlphaBIOS Setup** screen, select **Hard Disk Setup**, and press Enter.

3.   Put the Windows NT CD into the CD-ROM drive.

4.   Start **AlphaBIOS Setup**, select **Install Windows NT**, and press Enter.

5.   Follow the prompts to complete the installation. For more information on installing Windows NT, refer to the *Installation Guide* in your Windows NT software package.

5.8 Selecting the Version of Windows NT

Up to four versions of Windows NT can reside on the system at one time. Select the version that will be started from the Operating System Selection Setup screen.

Figure 5-24   Operating System Selections

```
                    Operating System Selection Setup


  Windows NT Server 4.0                    Primary Operating System
  Windows NT Server 5.0


    Boot Name:  Windows NT Server 4.0
    Boot File:  Disk 0, Partition 2    \os\winnt400\osloader.exe
    OS Path:    Disk 0, Partition 1    \WINNT400
    OS Options:


  INSERT=New      F6=Edit     F8=Primary     ESC=Discard Changes
  DEL=Delete      F7=Copy     F9=Validate    F10=Save Changes
```

```
                    Operating System Selection Setup


  Windows NT Server 5.0                    Primary Operating System
  Windows NT Server 4.0


    Boot Name:  Windows NT Server 5.0
    Boot File:  Disk 0, Partition 2    \os\winnt500\osloader.exe
    OS Path:    Disk 0, Partition 1    \WINNT500
    OS Options:


  INSERT=New      F6=Edit     F8=Primary     ESC=Discard Changes
  DEL=Delete      F7=Copy     F9=Validate    F10=Save Changes
```

PK-0723A-97

NOTE: The term "operating system selection," as it is used in this context, refers to a version of Windows NT. It does not pertain to the DIGITAL UNIX and OpenVMS operating systems.

Each operating system selection is a set of information for a version of Windows NT. It describes the disk and partition containing the OSLOADER.EXE file associated with a particular operating system installation, as well as the path to the operating system itself. In addition, the operating system selection contains any options passed to the operating system during boot.

By default, one operating system selection is displayed on the **AlphaBIOS Boot** screen for each operating system installed on the system. It is not normally necessary to modify operating system selections, as the Windows NT installation creates and modifies operating system selections as needed.

5.8.1 Designating a Primary Operating System

Figure 5-25   Primary Operating System

```
                          AlphaBIOS 5.29

Please select the operating system to start:

    Windows NT Server 4.0                              ❶
    Windows NT Server 5.0

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

```
                                        CMOS Setup


   Date:   Friday,    Nov      7    1997
   Time:   11:26:22


   Floppy Drive A: 3.5" 1.44 MB
   Floppy Drive B: None
   Keyboard:   U.S. 101-key keyboard


   Auto Start: Enabled                                ❷
   Auto Start Count:   30 Seconds
```

```
                    Operating System Selection Setup


Windows NT Server 4.0                  Primary Operating System   ❸
Windows NT Server 5.0
```

PK-0720D-97

Multiple versions of Windows NT can be installed at the same time. This can be very useful in a variety of circumstances—for example, when testing application compatibility across different versions of Windows NT.

Each time you install a separate version of Windows NT, a new operating system selection is created. Although you can start any of the installed versions of Windows NT, one of them must be the primary operating system. The version of Windows NT that you select as the primary operating system is the one that automatically starts if the **Auto Start** option is enabled in AlphaBIOS.

Figure 5-25 illustrates the relationship between multiple-operating-system installations, **Auto Start**, and the primary operating system.

❶   The primary operating system is listed first on the **Boot** screen.

❷   The primary operating system starts automatically if the **Auto Start** option is enabled in **CMOS Setup**.

❸   The primary operating system can be selected in the **Operating System Selection Setup** screen.

Follow this procedure to designate a new primary operating system:

1.   From the **AlphaBIOS Setup** screen select **Utilities**. In the selection box that is displayed, choose **OS Selection Setup**.

2.   The **Operating System Selections Setup** screen is displayed. Select the primary operating system from the list displayed.

5.8.2 Primary Operating System and the Auto Start Option

The process of setting up your operating system selections is similar to using an editor. You can make changes to your operating system selections and then either save your changes or exit without saving.

Figure 5-26   Operating System Selection Setup

```
                  Operating System Selection Setup


Windows NT Server 4.0   ❶                    Primary Operating System



   ❷  Boot Name:   Windows NT Server 4.0
   ❸  Boot File:   Disk 0, Partition 2    \os\winnt400\osloader.exe
   ❹  OS Path:     Disk 0, Partition 1    \WINNT400
   ❺  OS Options:

   ❻
   INSERT=New      F6=Edit     F8=Primary     ESC=Discard Changes
   DEL=Delete      F7=Copy     F9=Validate    F10=Save Changes
```

PK-0737C-97

AlphaBIOS boots Windows NT in two stages. The first stage involves reading an OS Loader program from a disk. The OS Loader program is named OSLOADER.EXE. The selection for the operating system must describe the path where AlphaBIOS will find the OS Loader program. The OS Loader program must be located on a partition formatted with the FAT file system.

Once the OS Loader program executes, it uses the services provided by AlphaBIOS to load the operating system components. After the operating system is loaded, the OS Loader starts execution of the operating system.

❶ Primary operating system — The OS that appears first on the **AlphaBIOS Boot** screen. It is also the version of the OS that automatically starts if **Auto Start** is selected. Any of the operating system selections can be the primary operating system.

❷ Boot name — Each boot name is associated with an operating system selection. Windows NT setup automatically creates a boot name each time you install the operating system. Because the boot name is only a name, you can modify it at any time without affecting the rest of the operating system selection. The boot name must have at least one character.

❸ Boot file — Describes the disk, partition, path, and name of the f ile that AlphaBIOS passes control to during the process of starting the operating system. This setting is created along with the operating system selection during Windows NT setup, and the user does not usually modify it. However, this setting can be modified if necessary. For example, a developer testing different versions of OSLOADER.EXE can store the different versions in different locations and modify this line to start the operating system with the different versions as needed.

During **Operating System Selection Setup**, you can select the disk and partition for the location of the boot file from a list of choices presented in a list box. To open a list of values for the field, press the Alt and down arrow keys together. When you select a boot file location and name, AlphaBIOS searches for the specified program on the specified partition. If the search fails, a warning is displayed, saying that the file does not exist. You are given the choice to continue with the changes anyway or to cancel the operation and fix the problem.

❹ OS path — Describes the disk, partition, and path to the operating system root directory for an operating system selection.

During **Operating System Selection Setup**, you can select the disk and partition for the location of the boot file from a list of choices presented in a list box. To open a list of possible values for the field, press the Alt and down arrow keys at the same time. When you select or enter an OS path, AlphaBIOS searches for the directory. If the search fails, a warning is displayed, saying that the directory does not exist. You are given the choice to continue with the changes anyway or to cancel the operation and fix the problem.

❺ OS options — Lists the startup parameters passed to the operating system for an operating system selection. One example of a startup parameter is whether to start the operating system in debug mode. By default, Windows NT does not add any entries to this field. This field can be modified.

❻ **Operating System Selection Setup** options — You can use the options listed at the bottom of the screen to edit operating system selections. When you edit an OS selection, the fields of the OS selection are validated when you exit the **OS Selection Setup** screen. Depending on the option you choose, one of the following dialog boxes will display.

New OS selection (Insert key) — Displays the **Insert New Operating System Selection** dialog box, with default values for the new OS selection already filled in. You can change these values as necessary.

Delete OS selection (Delete key) — Tags the currently selected OS selection for deletion. Although the OS selection is removed from the screen, it is not actually deleted until you save changes.

You can also delete all of your OS selections at once by pressing the Control and Delete keys at the same time while in the **Operating System Selection Setup** screen.

NOTE: If you delete all the OS selections, a dialog box is displayed informing you that no OS selections exist and offering three options. You can create a new OS selection, exit without saving changes, or exit and save changes.

Edit OS selection (F6 key) — Edit all values of an OS selection by selecting the OS selection to edit and pressing F6. A dialog box is displayed with current information. You can then edit the OS selection fields.

Copy OS selection (F7 key) — Create a new OS selection by using an existing OS selection as a template. Select the OS selection you want to copy and press F7. A dialog box is displayed with the values of your OS selection. You can then edit the OS selection fields. If you do not make any changes, a duplicate copy of the OS selection you copied is made.

Primary operating system (F8 key) — Sets the selected OS selection as the primary operating system. When you make an OS selection primary, it is displayed first on the **Operating System Selection Setup** screen with the text "Primary Operating System." In addition, the primary operating system is displayed first on the **AlphaBIOS Boot** screen and is automatically started if Auto Start is enabled.

Validate OS selection (F9 key) — Validates the fields in the currently selected OS selection. The validation routine checks that the OS loader file and OS directory fields contain valid paths and that the OSLOADER.EXE file exists in the directory specified. At the end of the validation, a dialog box is displayed describing the results of the validation. If there is an error in an OS selection, the validation routine displays a dialog box describing the component of the OS selection that is in error. At this point you can choose to edit the OS selection to correct the error, or delete the OS selection altogether.

You can also validate all OS selections at once by pressing the Control and F9 keys at the same time. All OS selections are validated in the order they are listed on screen.

Discard changes (Escape key) — Returns to the **AlphaBIOS Setup** screen without saving changes.

Save changes (F10 key) — Saves changes and returns to the **AlphaBIOS Setup** screen.

5.9 Switching from AlphaBIOS to SRM Console

It is necessary to switch to the SRM console to boot DIGITAL UNIX or OpenVMS or to run firmware-based diagnostics. To switch from AlphaBIOS to the SRM console, press the Rest button on the control panel, wait 5 seconds, and press the Halt button. Alternatively, select DIGITAL UNIX (SRM) or OpenVMS (SRM) from the Advanced CMOS Setup screen and reset the system.

Figure 5-27   Switching to the SRM Console

```
                          Advanced CMOS Setup              F1=Help

    PCI Parity Checking:  Disabled

    Power-up Memory Test:  Enabled

    AlphaBIOS Password Option:  Disabled

    Console Selection:  Digital UNIX Console (SRM)

    Power-up Memory Test Windows NT Console (AlphaBIOS)
                         OpenVMS Console (SRM)
                         Digital UNIX Console (SRM)




   Press ↑ or ↓ to select the firmware console that will be presented
   the next time the system is power-cycled.



   ESC=Discard Changes  F10=Save Changes
```

PK-0750B-96

PK-0750B-96

DIGITAL UNIX and OpenVMS are booted and firmware-based diagnostics are run from the SRM console. Follow this procedure to switch from AlphaBIOS to SRM:

1. Select **CMOS Setup** and press Enter.

2. In the **CMOS Setup** screen press F6. The **Advanced CMOS Setup** screen is displayed.

3. Select **DIGITAL UNIX console (SRM)** or **OpenVMS console (SRM)** and press F10.

4. The **CMOS Setup** screen is displayed. Press F10 to save the change.

5. Reset the system.

Alternatively, you can press the Reset button, wait 5 seconds, and then press the Halt button.

The method you choose depends on how you want the system to function on subsequent resets:

- If you make the selection in the **Advanced CMOS Setup** screen, the system will stop in the SRM console whenever it is powered up, reset, or initialized until the Windows NT operating system is again specified.

- If you press the Reset button, followed by the Halt button, no system setup information is changed, and AlphaBIOS will start the next time the system is powered up, reset, or initialized.

To return to the AlphaBIOS console, issue the command **alphabios** at the SRM console prompt.

5.10 Running Utility Programs

Maintenance programs such as RAID and EISA configuration utilities are run from the AlphaBIOS Utilities menu.

Figure 5-28   Run Maintenance Program Dialog Box
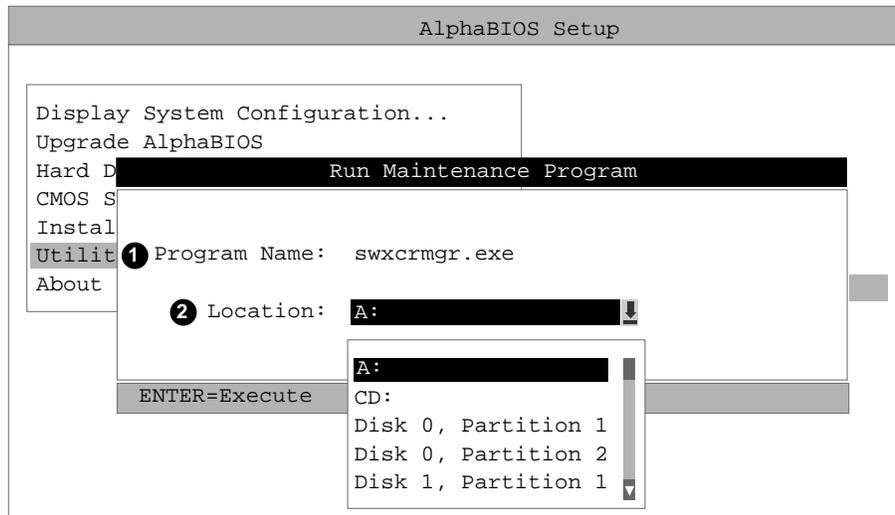
❶ Program name — The program to be run. It must be an executable program with an .EXE extension. It is not necessary to type the extension.

Programs run from AlphaBIOS must be written as ARC compatible images.

❷ Location — The location from which the program in the Program Name field will be run if no path is entered along with the program name. To display a list of available disks and partitions, press the Alt and down arrow keys at the same time. Once the list is displayed, use the arrow keys to cycle through the selections.

5.10.1 Running Utilities from a Graphics Monitor

1. Start AlphaBIOS Setup. If the system is in the SRM console, set the SRM **console** environment variable to **graphics** and issue the command **alphabios**.

2. From **AlphaBIOS Setup**, select **Utilities**, then select **Run Maintenance Program** from the submenu that is displayed, and press Enter. See Figure 5-29.

3. In the **Run Maintenance Program** dialog box (Figure 5-28), type the name of the program to be run in the Program Name field. Then Tab to the Location list box, and select the hard disk partition, floppy disk, or CD-ROM drive from which to run the program.

4. Press Enter to execute the program.

Figure 5-29   AlphaBIOS Utilities Menu

```
                          AlphaBIOS Setup                  F1=Help


     Display System Configuration...
     Upgrade AlphaBIOS
     Hard Disk Setup...
     CMOS Setup...
     Install Windows NT
     Utilities                          Run ECU from floppy...
     About AlphaBIOS...                 OS Selection Setup...
                                        Run Maintenance Program...



```
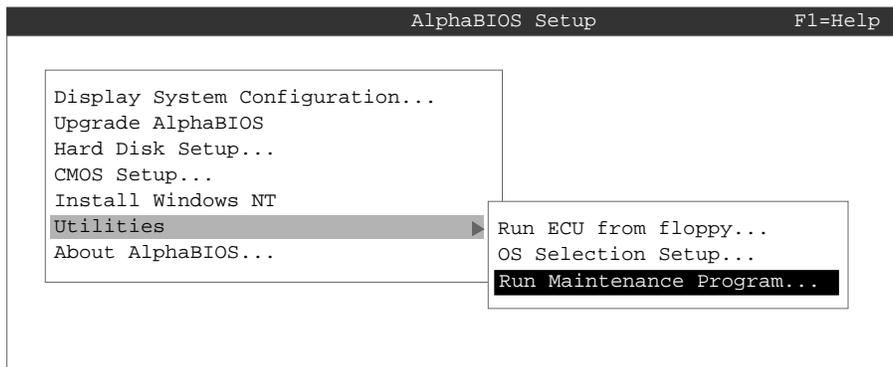
PK-0729-96

5.10.2 Running Utilities from a Serial Terminal

Utilities are run from a serial terminal in the same way as from a graphics monitor. The menus are the same, but some keys are different.

Table 5-1   AlphaBIOS Option Key Mapping

AlphaBIOS Key	VTxxx Key
F1	Ctrl/A
F2	Ctrl/B
F3	Ctrl/C
F4	Ctrl/D
F5	Ctrl/E
F6	Ctrl/F
F7	Ctrl/P
F8	Ctrl/R
F9	Ctrl/T
F10	Ctrl/U
Insert	Ctrl/V
Delete	Ctrl/W
Backspace	Ctrl/H
Escape	Ctrl/[

1. Start **AlphaBIOS Setup** by issuing the command **alphabios**.

2. Press F2 in the **AlphaBIOS Boot** screen.

3. From **AlphaBIOS Setup**, select **Utilities**, and select **Run Maintenance Program** from the submenu that is displayed. Press Enter.

4. In the **Run Maintenance Program** dialog box, type the name of the program to be run in the Program Name field. Then tab to the Location list box, and select the hard disk partition, floppy disk, or CD-ROM drive from which to run the program.

Press Enter to execute the program.

Chapter 6

Managing the System Remotely

This chapter describes how to manage the system from a remote location using the Remote Console Manager (RCM). You can use the RCM from a console terminal at a remote location. You can also use the RCM from the local console terminal.

Sections in this chapter are:

- RCM Overview
- First-Time Setup
- RCM Commands
- Dial-Out Alerts
- Using the RCM Switchpack
- Troubleshooting Guide
- Modem Dialog Details

6.1 RCM Overview

The remote console manager (RCM) monitors and controls the system remotely. The control logic resides on the system board.

The RCM is a separate console from the SRM and AlphaBIOS consoles. The RCM is run from a serial console terminal or terminal emulator. A command interface lets you to reset, halt, and power the system on or off, regardless of the state of the operating system or hardware. You can also use RCM to monitor system power and temperature.

You can invoke the RCM either remotely or through the local serial console terminal. Once in RCM command mode, you can enter commands to control and monitor the system. Only one RCM session can be active at a time.

- To connect to the RCM remotely, you dial in through a modem, enter a password, and then type an escape sequence that invokes RCM command mode. You must set up the modem before you can dial in remotely.

- To connect to the RCM locally, you type the escape sequence at the SRM console prompt on the local serial console terminal.

When you are not monitoring the system remotely, you can use the RCM dial-out alert feature. With dial-out alerts enabled, the RCM dials a paging service to alert you about a power failure within the system.

*CAUTION: Do not issue RCM commands until the system has powered up. If you enter certain RCM commands during power-up or reset, the system may hang. In that case you would have to disconnect the power cord at the power outlet. You can, however, use the RCM **halt** command during power-up to force a halt assertion. Refer to Section 3.12 for information on halt assertion.*

6.2 First-Time Setup

To set up the RCM to monitor a system remotely, connect the console terminal and modem to the ports at the back of the system, configure the modem port for dial-in, and dial in.

Figure 6-1   RCM Connections

PK-0906-97
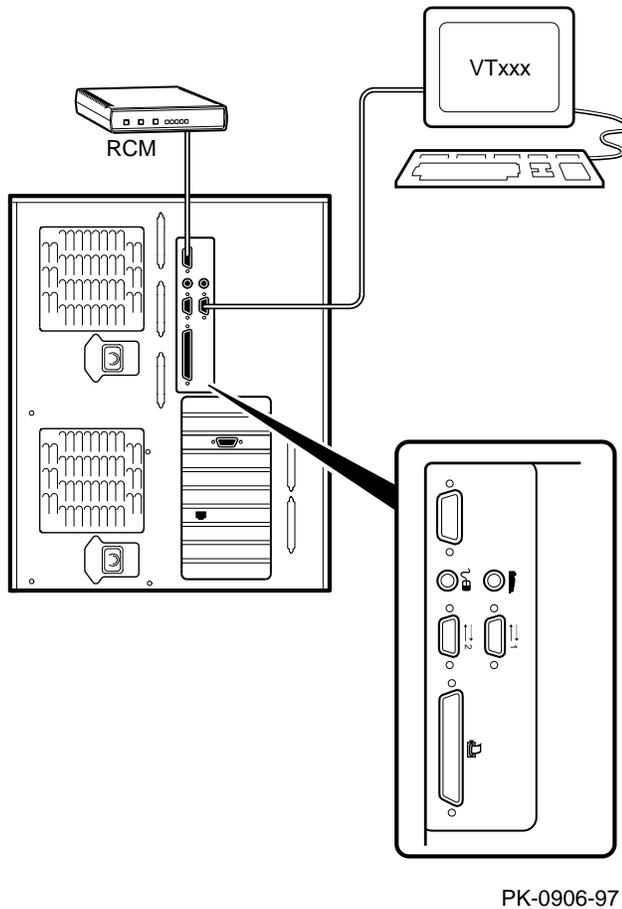
6.2.1 Configuring the Modem

The RCM requires a Hayes-compatible modem. The controls that the RCM sends to
the modem are acceptable to a wide selection of modems. After selecting the
modem, connect it and configure it.

Qualified Modems

The modems that have been tested and qualified with this system are:

- Motorola 3400 Lifestyle 28.8

- AT&T Dataport 14.4/FAX

- Hayes Smartmodem Optima 288 V-34/V.FC + FAX

Modem Configuration Procedure

1. Connect a Hayes-compatible modem to the RCM as shown in Figure 6-1, and
    power up the modem.

2. From the local serial console terminal, type the following escape sequence to
    invoke the RCM:

    P00>>> ^]^]rcm

    The character "^" is created by simultaneously holding down the Ctrl key and
    pressing the] key (right square bracket). The SRM prompt, RCM>, is
    displayed.

3. Use the **setpass** command to set a modem password.

4. Enable the modem port with the **enable** command.

5. Enter the **quit** command to leave the RCM.

6. You are now ready to dial in remotely.

6.2.2 Dialing In and Invoking RCM

To dial in to the RCM modem port, dial the modem, enter the modem password at the # prompt, and type the escape sequence. Use the hangup command to terminate the session.

A sample dial-in dialog would look similar to the following:

Example 6–1 Sample Remote Dial-In Dialog

```
ATQ0V1E1S0=0        ❶
OK
ATDT30167
CONNECT 9600
#                   ❷
RCM V2.0            ❸
RCM>
```

Dialing In and Invoking RCM

1. Dial the number for the modem connected to the modem port. See ❶ in Example 6–1 for an example.

    The RCM prompts for a password with a "#" character. See ❷.

2. Enter the password that you set with the **setpass** command.

    You have three tries to correctly enter the password. After three incorrect tries, the connection is terminated, and the modem is not answered again for 5 minutes. When you successfully enter the password, the RCM banner is displayed. See ❸. You are connected to the system COM1 port, and you have control of the SRM console.

    NOTE: At this point no one at the local terminal can perform any tasks except for typing the RCM escape sequence. The local terminal displays any SRM console output entered remotely.

3. Type the RCM escape sequence (not echoed).

    ```
    ^]^]rcm
    RCM>
    ```

    NOTE: From RCM command mode, you can change the escape sequence for invoking RCM, if desired. Use the setesc command to change the sequence. Be sure to record the new escape sequence.

4. To terminate the modem connection, enter the RCM **hangup** command.

```
RCM> hangup
```

If the modem connection is terminated without using the **hangup** command or if
the line is dropped due to phone-line problems, the RCM will detect carrier loss
and initiate an internal **hangup** command. If the modem link is idle for more
than 20 minutes, the RCM initiates an auto hangup.

*NOTE: Auto hangup can take a minute or more, and the local terminal is
locked out until the auto hangup is completed.*

6.2.3 Using RCM Locally

Use the default escape sequence to invoke the RCM mode locally for the first time.
You can invoke RCM from the SRM console, the operating system, or an
application. The RCM quit command reconnects the terminal to the system console
port.

1. To invoke the RCM locally, type the RCM escape sequence. See ❶ in Example
   6–2 for the default sequence.

   The escape sequence is not echoed on the terminal or sent to the system. At the
   RCM> prompt, you can enter RCM commands.

2. To exit RCM and reconnect to the system console port, enter the **quit** command.
   (see ❷). Press Return to get a prompt from the operating system or system
   console.

Example 6–2   Invoking and Leaving RCM Locally

```
P00>>> ^]^]rcm      ❶
RCM>

RCM> quit          ❷
Focus returned to COM port
```

6.3 RCM Commands

The RCM commands given in Table 6-1 are used to control and monitor a system remotely.

Table 6-1   RCM Command Summary

Command	Function
alert_clr	Clears alert flag, stopping dial-out alert cycle
alert_dis	Disables the dial-out alert function
alert_ena	Enables the dial-out alert function
disable	Disables remote access to the modem port
enable	Enables remote access to the modem port
halt	Halts the server. Emulates pressing the Halt button and immediately releasing it.
haltin	Causes a halt assertion. Emulates pressing the Halt button and holding it in.
haltout	Terminates a halt assertion created with **haltin**. Emulates releasing the Halt button after holding it in.
hangup	Terminates the modem connection
help or **?**	Displays the list of commands
poweroff	Turns off power. Emulates pressing the On/Off button to the off position.
poweron	Turns on power. Emulates pressing the On/Off button to the on position.
quit	Exits console mode and returns to system console port
reset	Resets the server. Emulates pressing the Reset button.
setesc	Changes the escape sequence for invoking command mode
setpass	Changes the modem access password
status	Displays system status and sensors

Command Conventions

- The commands are not case sensitive.
- A command must be entered in full.
- You can delete an incorrect command with the Backspace key before you press Enter.
- If you type a valid RCM command, followed by extra characters, and press Enter, the RCM accepts the correct command and ignores the extra characters.
- If you type an incorrect command and press Enter, the command fails with the message:

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

alert_clr

The **alert_clr** command clears an alert condition within the RCM. The alert enable condition remains active, and the RCM will again enter the alert condition if it detects a system power failure.

```
RCM>alert_clr
```

alert_dis

The **alert_dis** command disables RCM dial-out. It also clears any out standing alerts. Dial-out remains disabled until the **alert_enable** command is issued. See also the **enable** and **disable** commands.

```
RCM>alert_dis
```

alert_ena

The **alert_ena** command enables the RCM to automatically dial out when it detects a power failure within the system. The RCM repeats the dial-out alert at 30-minute intervals until the alert is cleared. Dial-out remains enabled until the **alert_disable** command or the **disable** command is issued. See also the **enable** and **disable** commands.

```
RCM>alert_ena
```

Two conditions must be met for the **alert_enable** command to work:

- A modem dial-out string must be entered from the system console.

- Remote access to the RCM modem port must be enabled with the **enable** command.

If the **alert_enable** command is entered when remote access is disabled, the following message is displayed:

```
*** error ***
```

disable

The **disable** command disables remote access to the RCM modem port. It also disables RCM dial-out.

```
RCM>disable
```

When the modem is disabled, it remains disabled until the **enable** command is issued. If a modem connection is in progress, entering the **disable** command terminates it.

*NOTE: If the modem has been disabled from the RCM switchpack on the motherboard, the **enable** command does not work. To enable the modem, reset the switch 2 (MODEM OFF) on the switchpack to OFF (enabled).See Section 6.5 for information on the switchpack.*

enable

The **enable** command enables remote access to the RCM modem port. It can take up to 10 seconds for the **enable** command to be executed.

```
RCM>enable
```

When the modem is enabled, it remains enabled until the **disable** command is issued.

The **enable** command can fail for the following reasons:

- No modem access password was set.

- The initialization string or the answer string might not be set properly. (See Section 6.7.)

- The modem is not connected or is not working properly.

- The modem has been disabled from the RCM switchpack. To enable the modem, reset switch 2 (MODEM OFF) on the switchpack to OFF (enabled *).

If the **enable** command fails, the following message is displayed:

```
*** ERROR - enable failed ***
```

hangup

The **hangup** command terminates the modem session. When this command is issued, the remote user is disconnected from the server. This command can be issued from either the local or remote console.

RCM>hangup

halt

The **halt** command halts the managed system. The **halt** command is equivalent to pressing the Halt button on the control panel and then immediately releasing it. The RCM firmware exits command mode and reconnects the user's terminal to the system COM1 serial port.

```
RCM>halt
Focus returned to COM port
```

The **halt** command can be used to force a halt assertion. See Section 3.12 for information on halt assertion.

*NOTE: If you are running Windows NT, the **halt** command has no effect.*

haltin

The **haltin** command halts a managed system and forces a halt assertion. The **haltin** command is equivalent to pressing the Halt button on the control panel and holding it in. This command can be used at any time after system power-up to allow you to perform system management tasks. See Section 3.12 for information on halt assertion.

*NOTE: If you are running Windows NT, the **haltin** command does not affect the operating system session, but it does cause a halt assertion.*

haltout

The **haltout** command terminates a halt assertion that was done with the **haltin** command. It is equivalent to releasing the Halt button on the control panel after holding it in (rather than pressing it once and releasing it immediately). This command can be used at any time after system power-up. See Section 3.12 for information on halt assertion.

help or ?

The **help** or **?** command displays the RCM firmware commands.

poweroff

The **poweroff** command requests the RCM to power off the system. The **poweroff** command is equivalent to pressing the On/Off button on the control panel to the off position.

```
RCM>poweroff
```

If the system is already powered off or if switch 3 (RPD DIS) on the switchpack has been set to the on setting (disabled), this command has no immediate effect.

To power the system on again after using the **poweroff** command, you must issue the **poweron** command.

If, for some reason, it is not possible to issue the **poweron** command, the local operator can start the system as follows:

1. Press the On/Off button to the off position and disconnect the power cord.

2. Reconnect the power cord and press the On/Off button to the on position.

poweron

The **poweron** command requests the RCM to power on the system. The **poweron** command is equivalent to pressing the On/Off button on the control panel to the on position. For the system power to come on, the following conditions must be met:

- AC power must be present at the power supply inputs.
- The On/Off button must be in the on position.
- All system interlocks must be set correctly.

The RCM exits command mode and reconnects the user's terminal to the system console port.

```
RCM>poweron
Focus returned to COM port
```

*NOTE: If the system is powered off with the On/Off button, the system will not power up. The RCM will not override the "off" state of the On/Off button. If the system is already powered on, the **poweron** command has no effect.*

quit

The **quit** command exits the user from command mode and reconnects the serial terminal to the system console port. The following message is displayed:

```
Focus returned to COM port
```

The next display depends on what the system was doing when the RCM was invoked. For example, if the RCM was invoked from the SRM console prompt, the console prompt will be displayed when you enter a carriage return. Or, if the RCM was invoked from the operating system prompt, the operating system prompt will be displayed when you enter a carriage return.

reset

The **reset** command requests the RCM to reset the hardware. The **reset** command is equivalent to pressing the Reset button on the control panel.

```
RCM>reset
Focus returned to COM port
```

The following events occur when the **reset** command is executed:

- The system restarts and the system console firmware reinitializes.

- The console exits RCM command mode and reconnects the serial terminal to the system COM1 serial port.

- The power-up messages are displayed, and then the console prompt is displayed or the operating system boot messages are displayed, depending on how the startup sequence has been defined.

setesc

The **setesc** command resets the default escape sequence for invoking RCM. The escape sequence can be any character string. A typical sequence consists of 2 or more characters, to a maximum of 15 characters. The escape sequence is stored in the module's on-board NVRAM.

NOTE: Be sure to record the new escape sequence. Although the factory defaults can be restored if you forget the escape sequence, this requires resetting the EN RCM switch on the RCM switchpack.

The following sample escape sequence consists of 5 iterations of the Ctrl key and the letter "o".

```
RCM>setesc
^o^o^o^o^o
RCM>
```

If the escape sequence entered exceeds 15 characters, the command fails with the message:

```
*** ERROR ***
```

When changing the default escape sequence, avoid using special characters that are used by the system's terminal emulator or applications.

Control characters are not echoed when entering the escape sequence. Use the **status** command to verify the complete escape sequence.

setpass

The **setpass** command allows the user to change the modem access password that is prompted for at the beginning of a modem session.

```
RCM>setpass
new pass>*********
RCM>
```

The maximum length for the password is 15 characters. If the password exceeds 15 characters, the command fails with the message:

```
***  ERROR  ***
```

Managing the System Remotely   **6-13**

The minimum password length is one character, followed by a carriage return. If only a carriage return is entered, the command fails with the message:

```
*** ERROR - illegal password ***
```

If you forget the password, you can enter a new password.

status

The **status** command displays the current state of the system sensors, as well as the current escape sequence and alarm information. The following is an example of the display.

```
RCM>status

Firmware Rev: V2.0
Escape Sequence: ^]^]RCM
Remote Access: ENABLE
Alerts: DISABLE
Alert Pending: NO
Temp (C): 26.0
RCM Power Control: ON
RCM Halt: Deasserted
External Power: ON
Server Power: ON

RCM>
```

The status fields are explained in Table 6-2.

Table 6-2   RCM Status Command Fields

Item	Description
Firmware Rev:	Revision of RCM firmware.
Escape Sequence:	Current escape sequence to invoke RCM.
Remote Access:	Modem remote access state. (ENABLE/DISABLE)
Alerts:	Alert dial-out state. (ENABLE/DISABLE)
Alert Pending:	Alert condition triggered. (YES/NO)
Temp (C):	Current system temperature in degrees Celsius.
RCM Power Control:	Current state of RCM system power control. (ON/OFF)
RCM Halt:	Asserted indicates that halt has been asserted with the **haltin** command. Deasserted indicates that halt has been deasserted with the **haltout** command or by cycling power with the On/Off button on the control panel. The RCM Halt: field does not report halts caused by pressing the Halt button.
External Power:	Current state of power to RCM. Always on.
Server Power:	Indicates whether power to the system is on or off.

6.4 Dial-Out Alerts

When you are not monitoring the system remotely, you can use the RCM dial-out feature to notify you of a power failure within the system.

When a dial-out alert is triggered, the RCM initializes the modem for dial-out, sends the dial-out string, hangs up the modem, and reconfigures the modem for dial-in. The modem must continue to be powered, and the phone line must remain active, for the dial-out alert feature to work. Also, if you are connected to the system remotely, the dial-out feature does not work.

Enabling Dial-Out Alerts

1. Enter the **set rcm_dialout** command, followed by a dial-out alert string, from the SRM console (see ❶ in Example 6–3). See the next topic for details on composing the modem dial-out string.

2. Invoke the RCM and enter the **enable** command to enable remote access dial-in. The RCM **status** command should display "Remote Access: Enable." See ❷.

3. Enter the **alert_ena** command to enable outgoing alerts. See ❸.

Example 6–3   Configuring the Modem for Dial-Out Alerts

```
P00>>> set rcm_dialout "ATDTstring#;"   ❶

RCM>enable
RCM>status
.
.
Remote Access:   Enable                 ❷
.
RCM>alert_ena                           ❸
```

6-16 User's Guide

Composing the Dial-Out String

Enter the **set rcm_dialout** command from the SRM console to compose the dial-out string. Use the **show** command to verify the string. See Example 6–4.

Example 6–4   Typical RCM Dial-Out Command

```
P00>>> set rcm_dialout "ATXDT9,15085553333,,,,,,5085553332#;"

P00>>> show rcm_dialout
rcm_dialout        ATXDT9,15085553333,,,,,,5085553332#;
```

The dial-out string has the following requirements:

- The string cannot exceed 47 characters.

- Enclose the entire string following the **set rcm_dialout** command in quotation marks.

- Enter the characters ATDT after the opening quotation marks. Do not mix case.

- Enter the character X after "AT" if the line to be used also carries voice mail.

- The valid characters for the dial-out string are the characters on a phone keypad: 0–9, *, and #. A comma (,) requests that the modem pause for 2 seconds, and a semicolon (;) is required to terminate the string.

The elements of the dial-out string are explained in Table 6-3.

Table 6-3 Elements of the Dial-Out String

ATXDT	AT = Attention X = Forces the modem to dial "blindly" (not look for a dial tone). Enter X if the dial-out line modifies its dial tone when used for services such as voice mail. D = Dial T = Tone (for touch-tone) , = Pause for 2 seconds
9,	In the example, "9" gets an outside line. Enter the number for an outside line if your system requires it.
15085553333	Dial the paging service.
,,,,,,	Pause for 12 seconds for paging service to answer
5085553332#	"Message," usually a call-back number for the paging service.
;	Return to command mode. Must be entered at end of string.

6.5 Using the RCM Switchpack

The RCM operating mode is controlled by a switchpack on the system board. Use the switches to enable or disable certain RCM functions, if desired.

Figure 6-2   Location of RCM Switchpack on System Board
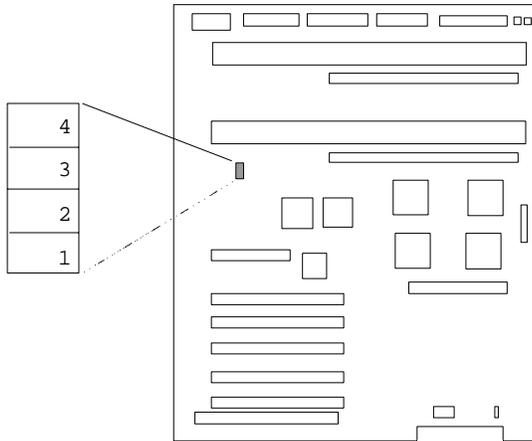
PKW0504C-97

Figure 6-3   RCM Switches (Factory Settings)
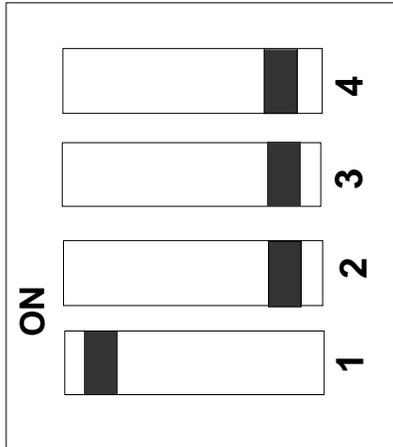
PKW0950-97

Switch	Name	Description
1	EN RCM	Enables or disables the RCM. The default is ON (RCM enabled). The OFF setting disables RCM.
2	MODEM OFF	Enables or disables the modem. The default is OFF (modem enabled).
3	RPD DIS	Enables or disables remote poweroff. The default is OFF (remote poweroff enabled).
4	SET DEF	Sets the RCM to the factory defaults. The default is OFF (reset to defaults disabled).

Uses of the Switchpack

You can use the RCM switchpack to change the RCM operating mode or disable the RCM altogether. The following are conditions when you might want to change the factory settings.

- Switch 1 (EN RCM)—Set this switch to OFF (disable) if you want to reset the baud rate of the COM1 port to a value other than the system default of 9600. You must disable RCM to select a baud rate other than 9600.

- Switch 2 (MODEM OFF)—Set this switch to ON (disable) if you want to prevent the use of the RCM for monitoring a system remotely. RCM commands can still be run from the local serial console terminal.

- Switch 3 (RPD DIS). Set this switch to ON (disable) if you want to disable the **poweroff** command. With **poweroff** disabled, the monitored system cannot be powered down from the RCM.

- Switch 4 (SET DEF). Set this switch to ON (enable) if you want to reset the RCM to the factory settings. See the section "Resetting the RCM to Factory Defaults."

Changing a Switch Setting

The RCM switches are numbered on the system board. The default positions are shown in Figure 6-3. To change a switch setting:

1. Turn off the system.

2. Unplug the AC power cords.

   NOTE: If you do not unplug the power cords, the new setting will not take effect when you power up the system.

3. Remove the system covers. See Section 7.2.

4. Locate the RCM switchpack on the system board and change the switch setting as desired.

5. Replace the system covers and plug in the power cords.

6. Power up the system to the SRM console prompt and type the escape sequence to enter RCM command mode, if desired.

Resetting the RCM to Factory Defaults

You can reset the RCM to factory settings, if desired. You would need to do this if you forgot the escape sequence for the RCM. Follow the steps below.

1. Turn off the system.

2. Unplug the AC power cords.

    NOTE: If you do not unplug the power cords, the reset will not take effect when you power up the system.

3. Remove the system covers. See Section 7.2.

4. Locate the RCM switchpack on the system board, and set switch 4 to ON.

5. Replace the system covers and plug in the power cords.

6. Power up the system to the SRM console prompt.

    Powering up with switch 4 set to ON resets the escape sequence, password, and modem enable states to the factory defaults.

7. Power down the system, unplug the AC power cords, and remove the system covers.

8. Set switch 4 to OFF.

9. Replace the system covers and plug in the power cords.

10. Power up the system to the SRM console prompt, and type the default escape sequence to invoke RCM command mode:

    `^]^]RCM`

11. Reset the modem password. Reset the escape sequence, if desired, as well as any other states.

6.6 Troubleshooting Guide

Table 6-4 is a list of possible causes and suggested solutions for symptoms you might see.

Table 6-4   RCM Troubleshooting

Symptom	Possible Cause	Suggested Solution
The local console terminal is not accepting input.	Cables not correctly installed.	Check external cable installation.
	Switch 1 on switchpack set to disable.	Set switch 1 to ON.
	Modem session was not terminated with the **hangup** command.	Wait several minutes for the local terminal to become active again.
	A remote RCM session is in progress, so the local console terminal is disabled.	Wait for the remote session to be completed.
The console terminal is displaying garbage.	System and terminal baud rate set incorrectly.	Disable RCM and set the system and terminal baud rates to 9600 baud.

Table 6-4   RCM Troubleshooting (continued)

Symptom	Possible Cause	Suggested Solution
RCM does not answer when the modem is called.	Modem cables may be incorrectly installed.	Check modem phone lines and connections.
		Enable remote access.
	RCM remote access is disabled.	Set password and enable remote access.
	RCM does not have a valid modem password set.	Set switch 1 to ON; switch 2 to OFF.
	Switch setting incorrect.	Enter **quit** on the local terminal.
	The local terminal is currently attached to the RCM.	Wait 30 seconds after powering up the system and RCM before attempting to dial in.
	On power-up, the RCM defers initializing the modem for 30 seconds to allow the modem to complete its internal diagnostics and initialization.	Enter **enable** command from RCM.
	Modem may have had power cycled since last being initialized or modem is not set up correctly.	
After the system and RCM are powered up, the COM port seems to hang briefly.	This delay is normal behavior.	Wait a few seconds for the COM port to start working.

Continued on next page

Table 6-4 RCM Troubleshooting (continued)

Symptom	Possible Cause	Suggested Solution
RCM installation is complete, but system does not power up.	RCM Power Control: is set to DISABLE.	Invoke RCM and issue the **poweron** command.
You reset the system to factory defaults, but the factory settings did not take effect.	AC power cords were not removed before you reset switch 4 on the RCM switchpack.	Refer to Section 6.5.
The remote user sees a "+++" string on the screen.	The modem is confirming whether the modem has really lost carrier. This occurs when the modem sees an idle time, followed by a "3," followed by a carriage return, with no subsequent traffic. If the modem is still connected, it will remain so.	This is normal behavior.
The message "unknown command" is displayed when the user enters a carriage return by itself.	The terminal or terminal emulator is including a linefeed character with the carriage return.	Change the terminal or terminal emulator setting so that "new line" is not selected.
Cannot enable modem or modem will not answer.	The modem is not configured correctly to work with the RCM.	Modify the modem initialization and/or answer string as described in Section 6.7.
	The modem has been disabled on the RCM switchpack.	Refer to Section 6.5.

6.7 Modem Dialog Details

This section is intended to help you reprogram your modem if necessary.

Default Initialization and Answer Strings

The modem initialization and answer command strings set at the factory for the RCM are:

Initialization string: `AT&F0EVS0=0S12=50<cr>`

Answer string   `ATXA<cr>`

NOTE: All modem commands must be terminated with a <cr> character (0x0d hex).

Modifying Initialization and Answer Strings

The initialization and answer strings are stored in the RCM's NVRAM. They come pre-programmed to support a wide selection of modems. With some modems, however, you may need to modify the initialization string, answer string, or both. The following SRM **set** and **show** commands are provided for this purpose.

To replace the initialization string:

```
P00>>>set rcm_init "new_init_string"
```

To replace the answer string:

```
P00>>>set rcm_answer "new_answer_string"
```

To display all the RCM strings that can be set by the user:

```
P00>>>show rcm*
rcm_answer  ATXA
rcm_dialout
rcm_init    AT&F0EVS0=0S12=50
P00>>>
```

Initialization String Substitutions

The following modems require modified initialization strings.

Modem Model	Initialization String
Motorola 3400 Lifestyle 28.8	`at&f0e0v0x0s0=2`
AT&T Dataport 14.4/FAX	`at&f0e0v0x0s0=2`
Hayes Smartmodem Optima 288 V-34/V.FC + FAX	`at&fe0v0x0s0=2`

Chapter 7

Installing Components

This chapter provides system component removal and replacement procedures. Sections include:

- Preparing to Install or Remove Components
- Top Cover and Side Panel Removal and Replacement
- Installing a CPU Module
- Installing a Memory DIMM Pair
- Installing a PCI or EISA/ISA Card
- Installing a Power Supply

CAUTION: Be sure to follow the appropriate antistatic precautions whenever handling internal components.

The following components should be installed by a service representative only:

- System board
- System backplane
- System fans
- Interlocking sensor switch
- Front bezel
- Operator control panel

For assistance in removing or installing these components, contact your service representative.

7.1 Preparing to Install or Remove Components

To prepare your system for installation and removal of components, you will need to assemble the required equipment, familiarize yourself with antistatic precautions, and remove the top and side panels of the system unit.

Required Equipment

You need the following equipment to perform the installation and removal procedures described in the following sections:

- Flat-head screwdriver
- Phillips screwdriver
- Antistatic wrist strap
- Replacement option board kit or device kit, if necessary

When handling internal system components, use an antistatic wrist strap to avoid damaging the components. Figure 7–1 shows how to attach the antistatic wrist strap to your wrist and to the system unit.

CAUTION: *Do not disconnect the power cords from the system or the power outlets.*
The power cord grounds the system unit, preventing damage to the
components.

Figure 7–1   Attaching the Antistatic Wrist Strap
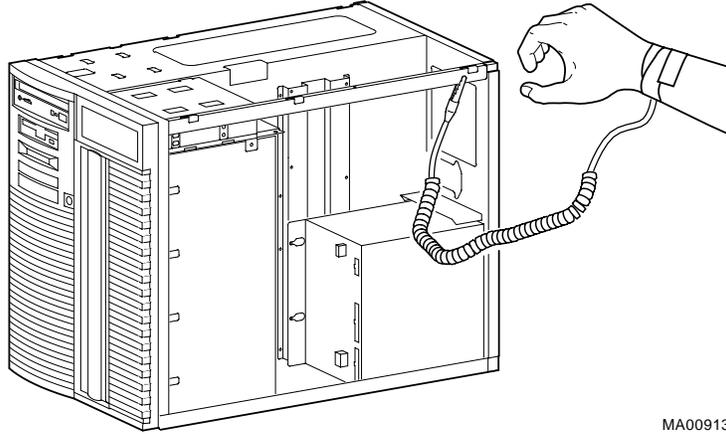
MA00913

7.2 Top Cover and Side Panel Removal and Replacement

Depending on the components involved, you may not need to remove all panels from the system unit. The top cover and left side panel must be removed to install or remove any internal component; the right side panel, to install or remove a power supply or a second SCSI controller.

Figure 7–2   Removing Top Cover and Side Panels

Top Cover
Release Latch

IP00205-97
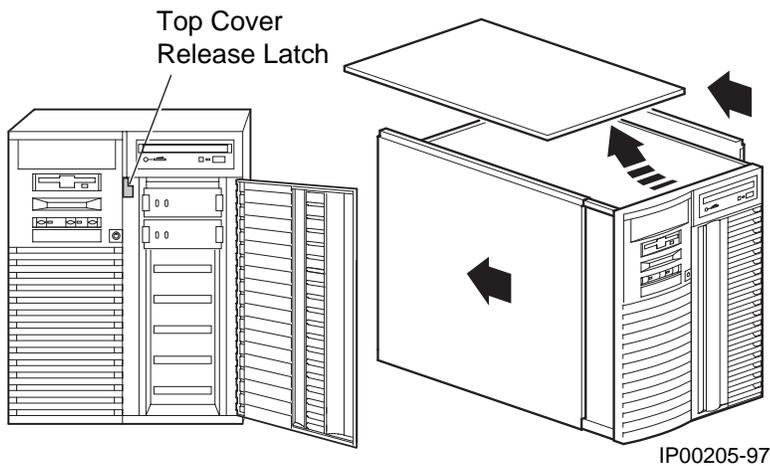
Top Cover and Side Panel Removal

CAUTION: Make sure the system unit On/Off button is in the "off" position before removing the system cover and panels.

To remove the top cover and side panels, refer to Figure 7–2 and follow these steps:

1. Shut down the operating system following the instructions listed in the operating system documentation.

2. Press the On/Off switches on all external options connected to the syst em to the off position.

3. Press the On/Off button on the system unit to the off position. An interlocking sensor switch inside the system unit will automatically turn off the system if you remove the top cover and have not turned off the system.

4. Turn the front door lock to the right to unlock the door.

5. Pull down the top cover release latch on the front of the system until it catches in the down position.

6. Grasp the finger groove at the rear of the top cover and pull the top c over straight back about 2 inches, and lift up on the cover.

7. Pull the top of the side panel back, then up and away from the unit to remove it. (Repeat this step for the other panel.)

Top Cover and Side Panel Replacement

Refer to Figure 7–3 to replace the top cover and side panels of the system unit.

Figure 7–3   Replacing Top Cover and Side Panels
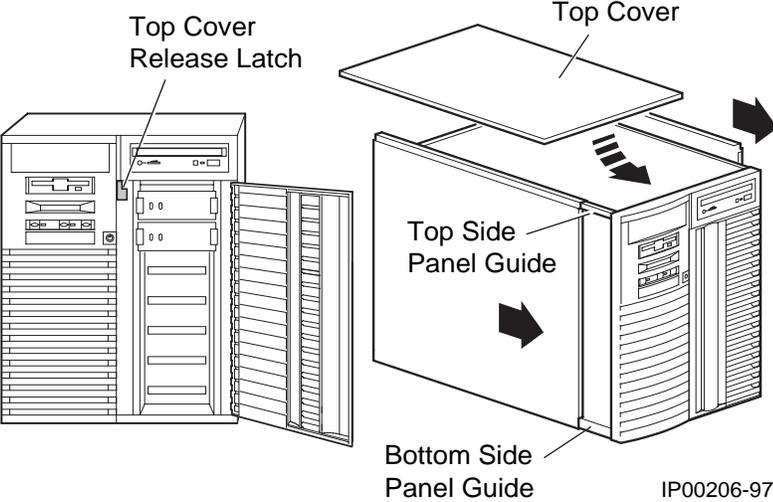
1. Align the guides on the top and bottom inside of the side panel with the lip of the system unit frame.

2. Tilt the side panel top toward the unit and lift the flange at the top of the panel over the system unit frame.

3. Slide the panel forward into position. (Repeat steps 1 to 3 for the other panel.)

4. Align the top cover with the top of the side panels and slide the cover gently onto the unit from the rear.

   *CAUTION: The top cover has a tab on the underside. Make sure to place this tab in the area **immediately behind** the microswitch. Do not hit the microswitch with the tab.*

4. Hold down the top cover release latch until the top cover is in place, and then release it to secure the cover and panels.

5. Lock the door to secure the top cover and side panels.

6. Set the On/Off switches on all external options connected to the system to the on position.

7. Press the On/Off button on the system unit to the on position.

7.3 Installing a CPU Module

Remove the top cover and left side panel to access the system board. After you have installed the new module, verify the installation by issuing the show cpu command from the SRM console or display the system bus configuration from the AlphaBIOS console.

Figure 7–4   CPU Module Slots
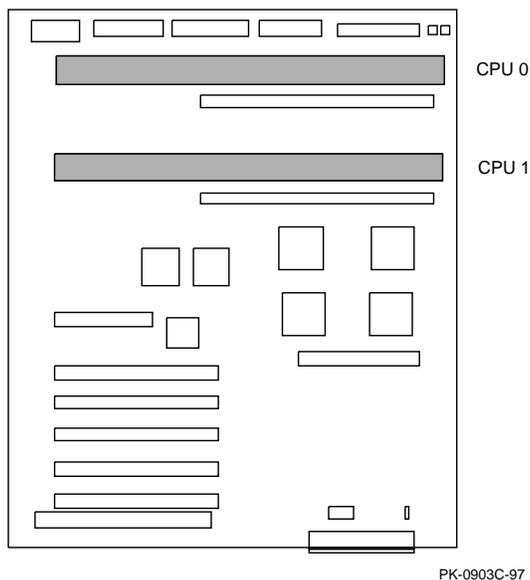
PK-0903C-97

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

CAUTION: Wear an antistatic wrist strap when working on this system.

Removal

NOTE: Before removing the CPU module, you must remove the memory card adjacent to it. See Section 7.4 for the memory card removal procedure.

1. Shut down the operating system and power down the system.
2. Remove the top cover and left side panel.

NOTE: Before removing the CPU card, you must remove the memory card adjacent to it. See Section 7.4 for the memory card removal procedure.

3. Unscrew the module handle that secures the outer edge of the CPU module (two retainer screws).

4. Release the card by extending the handle clip on each end outward.

5. Holding the outer edge of the card, gently pull it out of the slot.

Installation

1. Slide the CPU card into the system board slot.

2. Visually align the CPU card with the system board white connector and press the card into the connector. Check to see that the module handle clips are aligned with the slot notches.

3. Press inward on the handle clip on each end of the card until they catch in a closed position.

4. Secure the card by tightening the two captive retainer screws.

5. Replace the memory card.

Verification - DIGITAL UNIX and OpenVMS Systems

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

Verification - Windows NT Systems

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

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

Figure 7–5   CPU Module Installation
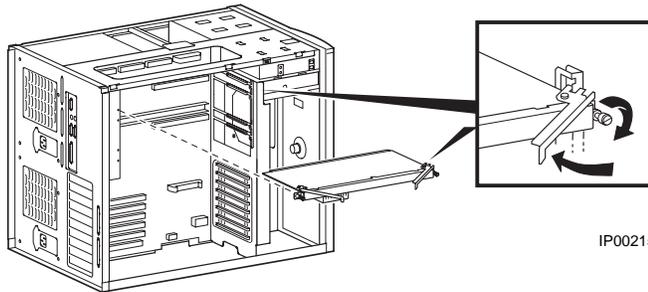
IP00215

7.4 Installing a Memory DIMM Pair

Remove the top cover and side panel to access the system board. After you have re-installed the memory card, verify the installation by issuing the show memory command from the SRM console or display the memory configuration from the AlphaBIOS console. Figure 7–6 shows the memory slots on the system board.

Configuration Rules

- There are two riser cards; one DIMM of each pair is installed in the same slot on each riser card (see Figure 7–7).
- Both DIMMs in a memory pair must be the same size, either 64 MB or 256 MB.
- The larger memory pair (256 MB) must be installed in riser card slot 0.
- Other memory pairs must be the same size or smaller than the first memory pair.
- Memory pairs must be installed in consecutive slots. The first pair goes into slot 0, the second pair into slot 1, and so on.
- If a 64-MB DIMM pair is installed in slot 0, the memory configuration cannot exceed 4 DIMM pairs.

Figure 7–6 Memory Slots on System Board

MEM L

MEM H
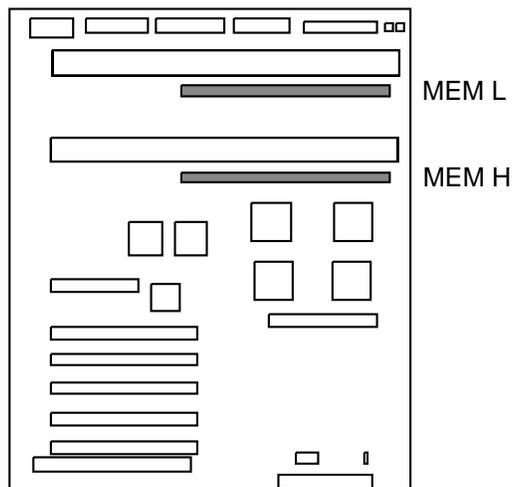
PK-0903D-97

Figure 7–7 Riser Card DIMM Slots
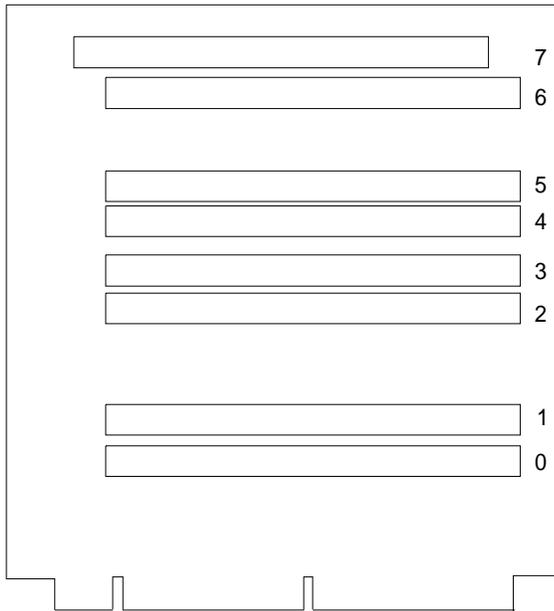
PKW0505A-97

Memory Card Removal and DIMM Installation

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

2. Remove the top cover and left side panel.

3. Unscrew the module handle that secures the outer edge of the memory card (two retainer screws). See Figure 7–8.

4. Holding the outer edge of the card, gently pull it out of the slot.

   CAUTION: Hold the memory card in your hands while installing DIMMs. Do not lay the card on a flat surface to install DIMMs. This may damage components on the other side of the memory card.

5. While holding the memory card, align the DIMM in the connector and press down. The connector levers close.

To install the memory card, reverse steps 1 through 4 in the removal procedure.

Figure 7–8 Memory Card Installation
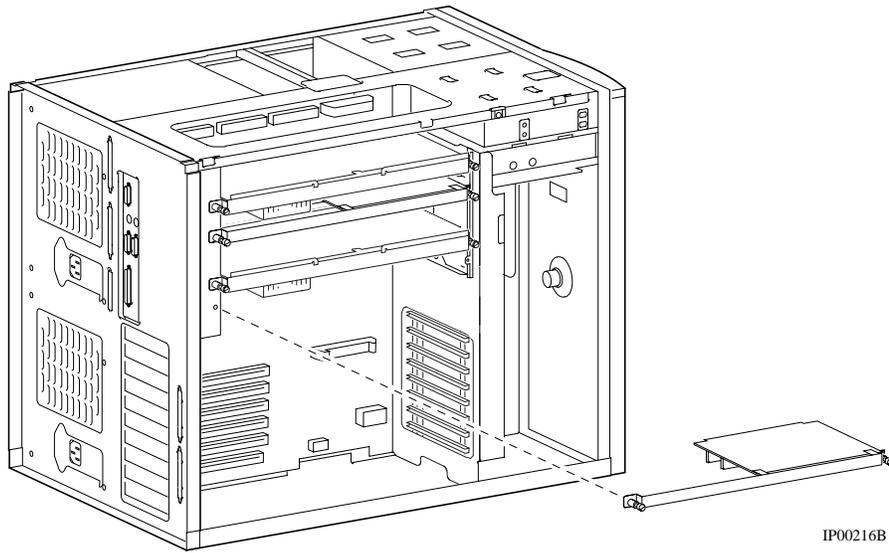
IP00216B

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

CAUTION: Wear an antistatic wrist strap when working on this system.

7.5 Installing a PCI or EISA/ISA Card

Remove the top cover and left side panel to access the system board. After you have installed the new module, verify the installation by issuing the show config and show device commands from the SRM console or display the PCI configuration from the AlphaBIOS console.

Configuration Rules

- SCSI controllers (KZPxx-xx) must be installed in PCI0.
- A graphics card can be installed in PCI0 slot 2.

Figure 7–9   PCI and EISA/ISA Slots
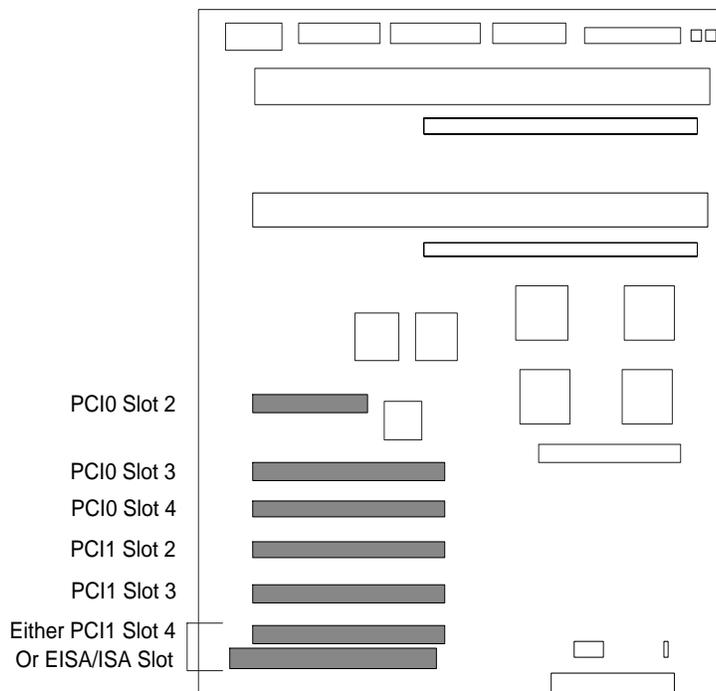
PK-0903E-97

Option Card Installation

1. Shut down the operating system and power down the system.
2. Remove the top cover and left side panel.
3. Select the I/O slot you wish to use and remove the screw that secures the slot cover to the chassis.
4. Install the option card by pressing it into the connector on the system board.
5. Secure the option card with the screw you removed from the slot cover.

Verification - DIGITAL UNIX and OpenVMS Systems

- Issue the SRM commands **show configuration** and **show device** to determine that the new option is listed.

Verification - Windows NT Systems

1. Start **AlphaBIOS Setup**, select **Display System Configuration**, and press Enter.
2. Using the arrow keys, select **PCI Configuration** to determine that the new option is listed.

Figure 7–10   PCI/ISA Card Installation

Slot Cover Screw

Option Card

IP00225
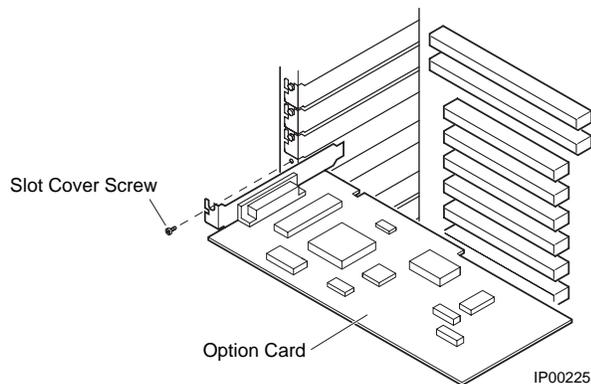
7.6 Installing a Power Supply

The following procedures describe how to install and remove a power supply. The procedures for installing and removing both power supplies are similar, unless otherwise noted.

Figure 7–11   Removing a Power Supply

IP00177
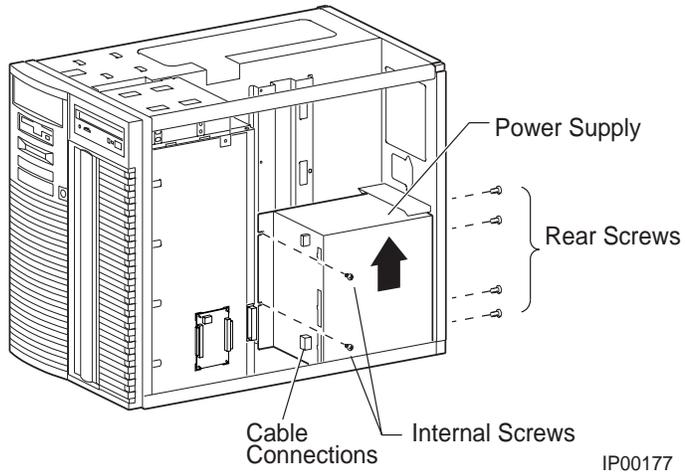
Power Supply Removal

1. Shut down the operating system and power down the system.
2. Remove the top cover and right side panel.
3. Disconnect all internal and external cables from the power supply.
4. Loosen the four screws that secure the power supply to the outside of the unit.
5. Loosen the two screws that secure the internal end of the power supply to the system unit.
6. Slide the power supply toward the internal side, lift it up off its tabs, and remove it through the top of the enclosure.

    NOTE: If you are removing the bottom power supply, you must remove the top power supply first.

Power Supply Installation

1. Insert the power supply through the top of the enclosure.
2. Slide the power supply over its tabs and toward the back panel so that it aligns with the screw holes in the system unit frame. The first supply slides directly into the lower position from the side of the system unit. The second power supply slides down from above the system unit and rests upon a ledge.
3. Tighten the two screws that secure the internal end of the power supply.
4. Tighten the four screws that secure the power supply to the outside of the unit. Do not use the screws that secured the protective shield.
5. Connect all internal and external cables to the power supply. If you are installing a second power supply, an additional internal cable is required for connecting the two supplies. Each power supply connects directly to an AC outlet.

Figure 7–12 Installing a Power Supply

Connecting Cable

Cable Connections Internal

Second Power Supply

Position for Second Power Supply

Rear Screws

Internal Screws
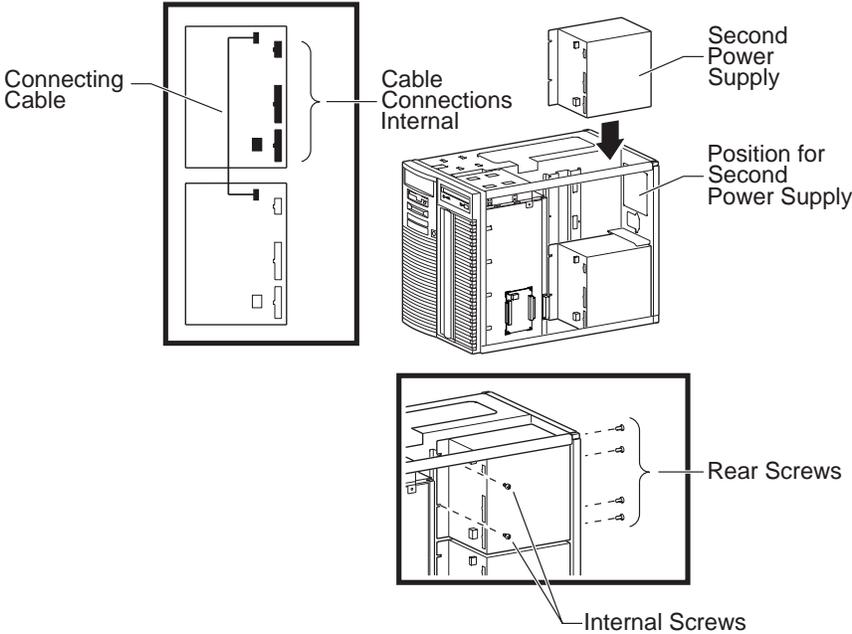
IP00176

Chapter 8

Troubleshooting

This chapter contains basic troubleshooting information. For more about troubleshooting, see the *AlphaServer 1200 Service Manual.*

Sections in this chapter are:

- System Does Not Power Up
- Control Panel Display Indicates an Error

8.1 System Does Not Power Up

When the system does not power up, the problem may be a disconnected power cord, a loose cable, a faulty fan, or a disengaged cover interlock.

Figure 8–1   Location of Cover Interlock

Cover Interlock
Switch
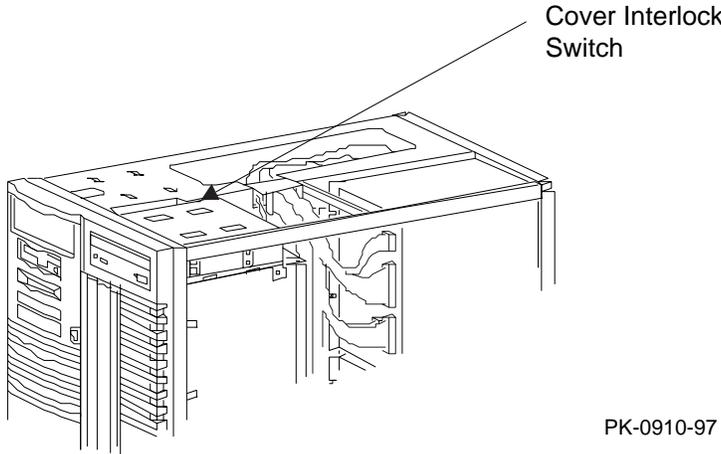
PK-0910-97

If the system does not power up

- Are the power cords plugged in?
- Are the cover and side panels closed and the interlock engaged?

  The cover interlock must be engaged to enable power-up. See Figure 8–1 for the interlock location.
- Is J2 engaged?
- Are the power supplies operating?
- Are the system fans operating?
- Are the CPU fans operating?
- Is RCM functional? If yes, has a **poweroff** command been issued?
- Are the system board LEDs on? See Figure 8–2.

Figure 8–2 System Board LEDs

System Board

IOD0 Pass
IOD1 Pass
Fans OK
Temp OK
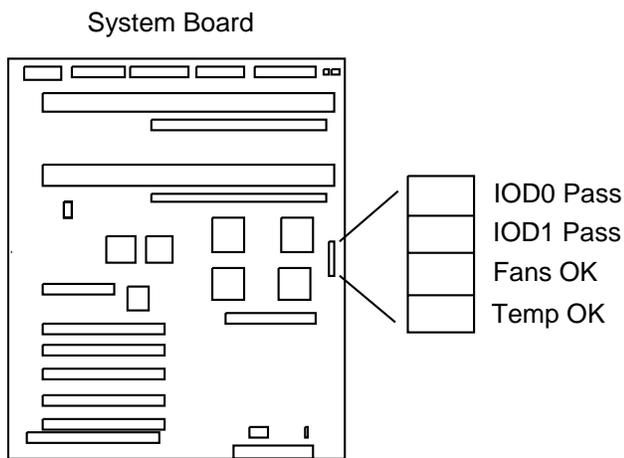
PK-0904-97

8.2 Control Panel Display Indicates an Error

The control panel display indicates the likely device causing the failure.

Figure 8–3   Control Panel Display
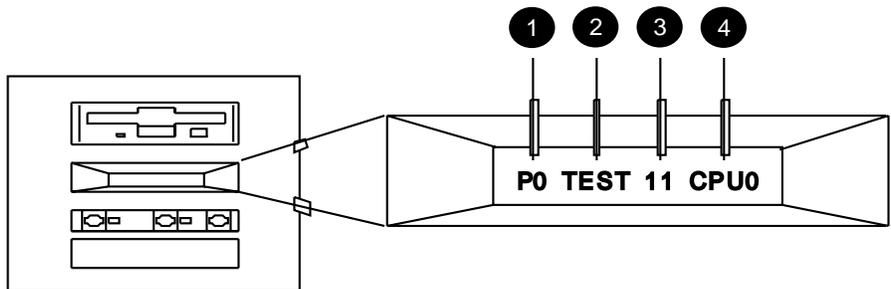
PK-0905-97

If the control panel display indicates an error

Check the failing device called out in the control panel display. See   Figure 8–3 and Table 8–1 for the location and meaning of the fields in the display.

Table 8–1   Control Panel Display

Field	Content	Display	Meaning
❶	CPU number	P0–P1	CPU reporting status
❷	Status	TEST	Tests are executing
		FAIL	Failure has been detected
		MCHK	Machine check has occurred
		INTR	Error interrupt has occurred
❸	Test number (for DIGITAL use only)		
❹	Suspected device	CPU0–1	CPU module number[1]
		MEM0–7 and L, H, or *	Memory pair number and low module, high module, or either[2]
		IOD0	Bridge to PCI bus 0[3]
		IOD1	Bridge to PCI bus 1[3]
		FROM0	Flash ROM[3]
		COMBO	COM controller[3]
		PCEB	PCI-to-EISA bridge[3]
		ESC	EISA system controller[3]
		NVRAM	Nonvolatile RAM[3]
		TOY	Real time clock[3]
		I8242	Keyboard and mouse controller[3]

[1] CPU module

[2] Memory module

[3] System board

Index