AlphaServer 2000/2100/2100A

Firmware Reference Guide

Order Number: EK–AXPFW–RM. C01

Digital Equipment Corporation
Maynard, Massachusetts

First Printing, October 1994
Revised, February 1995
Revised, December 1995

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

© Digital Equipment Corporation 1995. All Rights Reserved.

The following are trademarks of Digital Equipment Corporation: AlphaGeneration, AlphaServer, DEC, Digital, OpenVMS, ThinWire, VT, the AlphaGeneration logo, and the DIGITAL logo.

PostScript is a registered trademark of Adobe Systems, Inc. UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd. Microsoft Windows and Microsoft Windows NT are trademarks of Microsoft Corporation.

All other trademarks and registered trademarks are the property of their respective holders.

This document was prepared using VAX DOCUMENT Version 2.1.

Contents

Preface ix

1 Using the SRM Console Interface

Console Firmware Overview 1–2
Control and Keyboard Characters 1–8
SRM Console Commands 1–10
Entering Console Commands 1–12
Nvram Power-Up Script 1–13
arc 1–14
boot 1–15
cdp 1–17
continue 1–20
ecu 1–21
edit 1–22
help or man 1–24
init 1–25
man 1–27
more 1–28
set 1–29
set auto_action 1–33
set bootdef_dev 1–34
set boot_osflags 1–36
set bus_probe_algorithm 1–39
set com1_baud or set com2_baud 1–40
set console 1–41
set controlp 1–42
set cpu_enabled 1–43
set ew*0_mode 1–45
set ew*0_protocols 1–47
set host 1–49
set kbd_hardware_type 1–51
set language 1–52

set ocp_text 1–54
set os_type 1–55
set pci_parity 1–57
set pci_read_mult 1–58
set pk*0_fast 1–59
set pk*0_host_id 1–61
set pk*0_soft_term 1–62
set tga_sync_green 1–64
set tt_allow_login 1–65
show 1–66
show config 1–68
show device 1–75
show fru 1–81
show memory 1–85
show pal 1–86
show version 1–87
test 1–88

2 SRM Console Firmware Security Features

Security Features Overview 2–2
Security Commands 2–3
Secure Mode Functions 2–7

3 Using the ARC Console Interface

ARC Interface Introduction 3–5
Language Selection 3–7
Accessing the Boot Menu 3–10
Booting the Default ARC Boot Selection 3–12
Booting an Alternate Operating System 3–14
Running a Program 3–16
Accessing the Supplementary Menu 3–17
Accessing the Setup Menu 3–19
Displaying the Hardware Configuration 3–21
Setting the System Date and Time 3–25
Setting the Default System Partition 3–27
Configuring the System Floppy Drives and Keyboard Type 3–30
Setting the System to Boot Automatically 3–33
Editing the System Environment Variables 3–35
Default Environment Variables 3–38
Resetting the System to Factory Defaults 3–40
Getting Help 3–42

Saving the System Setup or Quitting the Setup Menu 3–45

4 Managing ARC Boot Selections

Accessing the Boot Selections Menu 4–2
Information on Boot Selection Variables 4–3
Adding a Boot Selection 4–4
Checking the Boot Selections for Errors 4–6
Changing a Boot Selection 4–8
Displaying the System Boot Selections 4–10
Changing the Default Boot Selection 4–12
Deleting a Boot Selection 4–14

Index

Examples

3–1 Boot Menu Example 3–11
3–2 Choosing the Boot Windows NT Menu Item 3–13
3–3 Booting an Alternate Operating System 3–15
3–4 Choosing the Alternate Operating System 3–15
3–5 Accessing the Supplementary Menu 3–18
3–6 Setup Menu 3–20
3–7 Sample Hardware Configuration Display 3–23
3–8 Set System Time Example 3–26
3–9 Setting the Default System Partition 3–28
3–10 Configuring the System Floppy Drives 3–31
3–11 Setting Autoboot 3–34
3–12 Creating a User-Defined Environment Variable 3–37
3–13 Selecting the Help Menu Item 3–43
3–14 Setup Menu Help Screen 3–44
4–1 Boot Selections Menu 4–2
4–2 Adding a Boot Selection 4–5
4–3 Checking the Boot Selections for Errors 4–7
4–4 Changing a Boot Selection 4–9
4–5 System Boot Selections Display 4–11
4–6 Changing the Default Boot Selection 4–13
4–7 Deleting a Boot Selection 4–14

Figures

1–1	Special Keyboard Operations	1–9
1–2	Device Naming Convention	1–75

Tables

1–1	Basic Console Commands	1–11
1–2	Environment Variables	1–31
1–3	Digital UNIX Flag Values	1–36
1–4	OpenVMS Boot Flag Settings	1–37
1–5	Hex Digit Values for CPUs	1–44
1–6	Device Naming Convention: AlphaServer 2100A	1–76
1–7	Device Naming Convention: AlphaServer 2000 and 2100	1–77
3–1	Accessing the Boot Menu	3–10
3–2	Booting the Default Boot Selection	3–12
3–3	Booting an Alternate Operating System	3–14
3–4	Running a Program	3–16
3–5	Accessing the Supplementary Menu	3–17
3–6	Accessing the Setup Menu	3–19
3–7	Displaying the Hardware Configuration	3–21
3–8	Device Names	3–22
3–9	Setting the System Date and Time	3–25
3–10	Setting the Default System Partition	3–27
3–11	Effects of Setting the Default System Partition	3–29
3–12	Configuring the Floppy Drives and Keyboard	3–30
3–13	Effects of Configuring the System Floppy Drives	3–32
3–14	Setting the System to Boot Automatically	3–33
3–15	Effects of Setting the System to Boot Automatically	3–34
3–16	Creating or Editing User-Defined Environment Variables	3–36
3–17	ARC Firmware Environment Variables	3–38
3–18	Resetting to Factory Defaults	3–40
3–19	Displaying the Setup Menu Help Screen	3–42
3–20	Saving the System Setup	3–45
3–21	Quitting the Setup Menu	3–45
4–1	Accessing the Boot Selections Menu	4–2
4–2	ARC Boot Selection Variables	4–3

4–3	Adding a Boot Selection	4–4
4–4	Checking the Boot Selections for Errors	4–6
4–5	Changing a Boot Selection	4–8
4–6	Displaying the System Boot Selections	4–10
4–7	Changing the Default Boot Selection	4–12
4–8	Deleting a Boot Selection	4–14

Preface

Purpose of This Guide

This guide contains firmware reference information about SRM console commands and ARC firmware menus for the AlphaServer 2000, 2100, and 2100A systems.

Who Should Use This Guide

This guide is for system managers and others who perform system management tasks that require interaction between the user, the system firmware, and the operating system.

Structure of This Guide

This guide contains the following chapters:

- Chapter 1 describes how to use the SRM command line interface and provides reference information for the SRM console firmware commands and environment variables.
- Chapter 2 explains how to implement console security features on the system.
- Chapter 3 describes how to use the ARC menu interface for Windows NT to boot the system, change the system setup, and carry out other system management functions.
- Chapter 4 describes how to change boot selections in the ARC menu interface.

For More Information

You may find it useful to have your operating system documentation and your system hardware documentation on hand for use with this guide.

You can download up-to-date files and information for managing AlphaServer 2000, 2100, and 2100A systems from the Internet at the following locations. The information includes firmware updates, software patches, lists of supported options, and more.

- ftp://ftp.digital.com/pub/Digital/Alpha/systems/
- http://www.service.digital.com/alpha/server/

Firmware Updates

An Alpha Firmware CD-ROM is shipped on an ‘‘as released’’ basis with the Digital UNIX, OpenVMS, and Windows NT operating systems. The Alpha firmware files can also be downloaded from the Internet as follows:

- ftp://ftp.digital.com/pub/Digital/Alpha/firmware/
- http://www.service.digital.com/alpha/server/firmware/

New versions of firmware released between shipments of the Alpha Firmware CD-ROM are available in an interim directory:

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

Conventions

The following table lists conventions used in this guide.

Convention	Meaning
Return	A key name enclosed in a box indicates that you press that key.
Ctrl/*x*	Ctrl/*x* indicates that you hold down the Ctrl key while you press another key, indicated here by *x*. In examples, this key combination is enclosed in a box, for example, Ctrl/C.
lowercase	Lowercase letters in commands indicate that commands can be entered in uppercase or lowercase.
Warning	Warnings contain information to prevent personal injury.
Caution	Cautions provide information to prevent damage to equipment or software.
[ ]	In command format descriptions, brackets indicate optional elements.
console command abbreviations	Console command abbreviations must be entered exactly as shown.
`boot`	Console and operating system commands are shown in this special typeface.
italic type	Italic type in console command sections indicates a variable.
< >	In console mode online help, angle brackets enclose a placeholder for which you must specify a value.
{ }	In command descriptions, braces containing items separated by commas imply mutually exclusive items.
❶ ❷ ❸ . . .	In figures, a numbered callout labels a part of the figure. In text, it refers to a labeled part of a corresponding figure.

1

Using the SRM Console Interface

Introduction

On AlphaServer systems, control of the system hardware is provided by a console subsystem. The console subsystem contains firmware code (software code embedded in hardware) that facilitates interaction between the user, the hardware, and the operating system.

Because the AlphaServer 2000, 2100, and 2100A systems are designed to support multiple operating systems, these servers offer two separate operator interfaces:

- A command line interface called the SRM interface for Digital UNIX and OpenVMS systems. The SRM interface provides bootstrap and other services as defined in the *Alpha System Reference Manual* (SRM) specification.
- A menu interface called the ARC interface for the Windows NT operating system. The ARC interface provides bootstrap and other services as defined in the *Windows NT Portable Boot Loader Specification*.

In This Chapter

This chapter explains how to use the SRM command line interface to carry out operations such as testing the hardware and bootstrapping the system software. The command line interface will be familiar to users of Digital's VAX line of computers.

Console Firmware Overview

The AlphaServer 2000, 2100, and 2100A systems contain firmware code that provides service functions, including power-up diagnostics and initialization, console interfaces, and operating system boot and restart.

Firmware Services

At power-up, the firmware tests and initializes the base system. The firmware provides a set of built-in drivers to permit generic access to all the major physical components in the system.

The initialization code is run any time the system is reset. During initialization the base system hardware is tested and configured. All processor caches are set to a consistent state, memory is sized and configured, and the I/O subsystem is reset and configured.

After initialization the firmware starts up its own internal set of class and port drivers. The drivers are used by the console, diagnostics, bootstrap, and the operating system callbacks. This arrangement simplifies the addition of new device support, since it typically requires only the addition of port level drivers.

System self-tests are performed at power-up. These tests include memory diagnostics, individual device test-directed diagnostics executed within a driver's initialization routine, and a concurrent system exerciser. This power-up stream can be customized using both built-in and user scripts.

Operating System Support

The system firmware provides support for booting the Digital UNIX, OpenVMS, and Microsoft Windows NT operating systems.

- Digital UNIX and OpenVMS are booted from the SRM firmware. The console terminal used as the operator interface to Digital UNIX and OpenVMS can be either a serial terminal (VT320 or higher, or equivalent) or a VGA-compatible graphics monitor.
- Microsoft Windows NT is booted from the ARC firmware. The console terminal used as the operator interface to Windows NT must be a VGA-compatible graphics monitor.

SRM Command Line Interface

Systems running Digital UNIX or OpenVMS access the SRM firmware through a command line interface (CLI). The CLI is a UNIX style shell that provides a set of commands and operators, as well as a scripting facility. It allows you to configure and test the system and boot the operating system.

To access the SRM console, power up the system as described in the system Owner's Guide.

To return to the SRM console after an operating system session, shut down the operating system as described in your operating system documentation.

The SRM console prompt is `P00>>>`.

SRM-Only Tasks

There are several system management tasks that can be performed only from the SRM console command line interface:

- To test the system, use the SRM `test` command.
- To examine system options for errors, use the `show fru` command.
- To set or change certain environment variables, use the SRM `set` command. The environment variables you can change with the `set` command are listed alphabetically in the reference section of this chapter.

ARC Utilities

Although the ARC console firmware is used primarily to configure and boot the system, it is also used to run ARC-compliant utilities. The ARC console described in Chapter 3 and Chapter 4.

Two ARC utilities commonly used for configuring system options are the EISA Configuration Utility (ECU) and the RAID Standalone Configuration Utility.

EISA Configuration Utility

The ECU is used for configuring EISA options and some ISA options. You run the ECU under the following circumstances:

- When you add or move EISA options and some ISA options.
- When you remove EISA options or some ISA options from a system running ARC firmware lower than Version 4.38.

The ECU is run from diskette. Two diskettes are supplied with your hardware system shipment, one for Digital UNIX and OpenVMS and one for Windows NT. To run the ECU, refer to the instructions in the system Owner's Guide.

RAID Standalone Configuration Utility

The RAID Standalone Configuration Utility is used on systems that have optional StorageWorks RAID Array Subsystem controllers installed. You run the RAID Standalone Configuration Utility to set up the disk drives and logical units. For instructions, refer to *StorageWorks RAID Array 200 Subsystems Controller Installation and Standalone Configuration Utility User's Guide*, included in your StorageWorks RAID kit.

Serial-Line Support for ECU and RAID Utility

You can run the ECU and the RAID Standalone Configuration Utility from either a VGA-compatible graphics monitor or from a serial terminal (VT320 or higher, or equivalent). To use a serial terminal with these utilities, set up your terminal as follows:

1. Invoke the terminal setup utility as described in the documentation for your serial terminal, and change settings as follows:
   - From the General menu, set the terminal mode to VTxxx mode, 8-bit controls.
   - From the Comm menu, set the character format to 8 bit, no parity, and set receive XOFF to 128 or greater.
2. Enter the following commands at the SRM console prompt to set the console terminal to receive input in serial mode.

```
P00>>> set console serial
P00>>> init
   .
   .
   .                    !Now switch to the serial terminal.
P00>>> show console
console      serial
```

Switching Between Interfaces

To complete some procedures, it is necessary to switch from one console interface to the other.

- Several commands, including the `test` command, are run under the SRM interface.
- The EISA Configuration Utility (ECU) and the RAID Standalone Configuration Utility are run under the ARC interface.

Switching from SRM to ARC

To switch from the SRM console to the ARC console, enter the `arc` command at the SRM console prompt `P00>>>`.

Switching from ARC to SRM

Switch from the ARC console to the SRM console as follows:

1. From the Boot menu, select `Supplementary menu`.
2. From the Supplementary menu, select `Set up the system . . . .`
3. From the Setup menu, select `Switch to OpenVMS or Digital UNIX console.`
4. Select your operating system console, then select the Setup menu and press Enter.
5. When the `Power-cycle the system to implement the change` message is displayed, press the Reset button.

   Once the console firmware is loaded and the system is initialized, the SRM console prompt, `P00>>>`, is displayed.

Control and Keyboard Characters

Control Characters

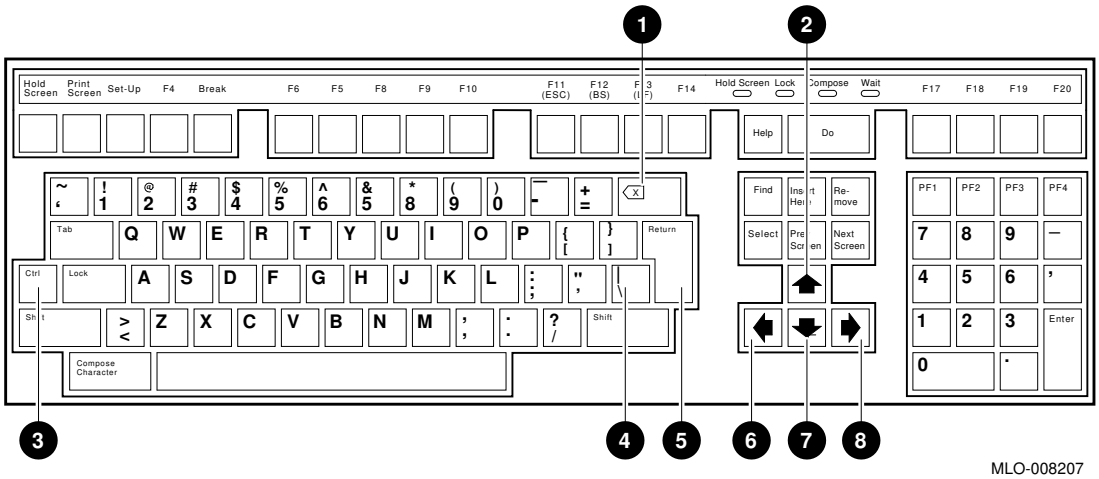
Enter control characters by holding down the key labeled Ctrl while pressing another key. You can enter the following control characters in console mode:

Character	Function
Ctrl/A	Toggles between insertion and overstrike mode so you can edit text on the current command line. Default mode is insert.
Ctrl/C	Interrupts a command process and returns control to the console command line.
Ctrl/E	Moves the cursor to the end of the line.
Ctrl/H	Moves the cursor to the beginning of the line.
Ctrl/O	Suppresses output to console terminal until you enter Ctrl/O again. Output is also reenabled when the console prompts for a command, issues an error message, or enters operating system mode.
Ctrl/P	Halts the OpenVMS operating system.
Ctrl/S	Suspends output to the console terminal until you enter Ctrl/Q.
Ctrl/Q	Resumes output to the console terminal that you suspended with Ctrl/S.
Ctrl/U	Deletes the entire line. Line deletion is followed with a carriage return, line feed, and a new prompt.
Ctrl/R	Redisplays the current line, omitting deleted characters.

Keyboard Characters

Figure 1–1 shows the keyboard characters that operate in console mode. The location of keyboard characters may vary depending on your keyboard. You can, however, enter the following keyboard characters in console mode, regardless of your keyboard type.

Figure 1–1 Special Keyboard Operations



Callout	Key	Function
❶	<X	Deletes the last character entered. With a hardcopy terminal, < X is echoed with \ followed by the character being deleted. If you delete several characters consecutively, the system echoes with \, the deleted characters, followed by another \ at the end of the series.
❷	↑	Recalls up to 32 previous commands.
❸	Ctrl	Begins a control character.
❹	\ (Backslash)	Extends a command onto the next line. Must be the last character on the line to be continued.
❺	Return	Enters a command on the command line. The cursor need not be at the end of the command line.
❻	←	Moves the cursor left one position.
❼	↓	Reverses the order of recalled commands after using ↑.
❽	→	Moves the cursor right one position.

SRM Console Commands

Levels of Commands

There are two levels of SRM console commands:

- Basic
- Advanced

Basic commands are the commands commonly used to perform system management tasks such as booting the operating system or setting environment variables. These commands are documented in alphabetical order in this chapter. Table 1–1 briefly describes the basic console commands.

Advanced console commands generally are used only by the power-up and test scripts. The reserved commands appear in the `help` or `man` display, but are not documented in this manual because they are not needed for typical system management tasks.

Table 1–1 Basic Console Commands

Command	Syntax	Description
arc	arc	Invokes the ARC console firmware.
boot	boot [-flags [*flags_value*]] [-halt] [*boot_device*]	Boots the operating system software.
cdp	cdp [-{a,i,n,o,u}] [-sn] [-sa *allclass*] [-su *unitnum*] [*dssi_device*]	Configures DSSI drive IDs.
continue	continue [*processor*]	Resumes program execution.
ecu	ecu	Invokes the EISA Configuration Utility.
edit	edit *file*	Invokes the console line editor on a RAM file or on the nvram file.
help	help [*command . . .*]	Displays online help using console commands. Synonymous with man.
init	init	Initializes the system.
man	man [*command . . .*]	Displays online help using console commands. Synonymous with help.
more	more *file*	Displays a RAM file that you specify a screen at a time.
set	set [-default] *envar val*	Sets an environment variable.
set host	set host [-dup] [-task *task_name*] *device*	Connects to a MSCP DUP server on a DSSI device.
show	show [*envar*] [{config, device, fru, memory, pal, version}]	Displays the value of an environment variable or displays configuration information.
test	test	Runs system diagnostics.

Entering Console Commands

Console Command Format

Use the following format when entering console commands. To specify an argument or value, precede it with a space. To specify an option, precede the option with a space and a hyphen.

```
P00>>> command [-options [value]] [arguments. . .]
```

For information about entering a specific console command, refer to the console command reference pages, in the next section of this chapter.

Online Help

Once the system is running in console mode, you can get online information about console commands by entering `help` or `man` at the console terminal.

Nvram Power-Up Script

The system comes with a special nonvolatile file, named ‘‘nvram,’’ that is stored in EEROM in the I/O subsystem. Nvram is a user-defined power-up script that is always invoked during the power-up sequence. A script is a group of commands to be executed.

The nvram script is initially empty. You can customize ‘‘nvram’’ to include any actions you want the system to execute at power-up. The nvram script can also be edited to apply patches, if needed.

Console Editor

You customize the nvram script using the `edit` command. `Edit` works much like the line editor for the BASIC computer language. With `edit,` lines of a file may be added, inserted, or deleted. For information on creating an nvram script, refer to the ‘‘edit’’ command section in this chapter.

arc

Synopsis

Invoke the ARC firmware.

arc

Description

Entering the `arc` command from the console prompt loads the ARC firmware and boots the ARC console. You do not need to restart the system in order to load the ARC firmware.

Example

In the following example, the ARC console is invoked from the SRM console.

```
P00>>> arc
```

boot

Synopsis

Bootstrap the system.

boot [-flags [*flags_value*]] [-halt] [*boot_device*]

Description

Initializes the processor, loads a program image from the specified boot device, and transfers control to that image.

If you specify a list of devices, a bootstrap is attempted from each device in order. Then control passes to the first successfully booted image. In a list, always enter network devices last, since network bootstraps only terminate if a fatal error occurs or an image is successfully loaded.

The -flags option passes additional information to the operating system about the boot that you are requesting.

You can execute the `boot` command by entering `boot` or `b`, followed by Return. When you enter `boot` without options, the environment variables shown in parentheses below provide default values for the following information:

- Boot device (bootdef_dev)
- Boot flags (boot_osflags)

If you specify the boot device or boot flags on the command line, the current default value is overridden for the current boot request. However, the corresponding environment variable is not changed.

Arguments

Argument	Description
boot_device	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. Use the `set bootdef_dev` command to define the default boot device.

boot

Options

Options	Description
-flags [*flags_value*]	Specifies additional information to the loaded image or operating system. The information is operating system specific. Refer to the `set boot_osflags` command for a list of possible settings and their meanings. The default boot flag setting is null. Use the `set boot_osflags` command to change the default boot flag setting.
-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 when this qualifier is present. Transfer control to the image by entering the `continue` command.

Examples

In the following example, an OpenVMS system boots from the SCSI disk, dka0. If bootable software cannot be found on dka0, the system attempts to boot from Ethernet port ewa0. Boot flag settings are 0 and 1.

```
P00>>> boot -flags 0,1 dka0,ewa0
```

In the following example, a Digital UNIX system boots from the SCSI disk, dka0, using boot flag setting i.

```
P00>>> boot -flags i dka0
```

In the following example, the system boots from the SCSI disk, dka0, but remains in console mode. Subsequently, you can enter the `continue` command to transfer control to the operating system.

```
P00>>> boot -halt dka0
```

In the following example, the system boots from the default boot device. The console program returns an error message if a default boot device has not been set.

```
P00>>> boot
```

cdp

Synopsis

Configure DSSI parameters.

cdp [-{a,i,n,o,u}] [-sn] [-sa *allclass*] [-su *unitnum*] [*dssi_device*]

Description

The `cdp` command allows you to modify DSSI device parameters from the console terminal without connecting to a node's DUP server. The parameters that are modified are the DUP task parameters: NODENAME, ALLCLASS, and UNITNUM.

If you enter `cdp` without an option or target device, the system displays parameters for all DSSI devices in the system.

Note

Some DSSI controllers do not support the `cdp` command. In these cases, the `set host -dup` command, described in this chapter, can be used to configure DSSI parameters.

Arguments

Argument	Description
dssi_device	Name of the DSSI device or DSSI controller. Only the parameters for this device or the devices on this controller will be modified.

Options

Option	Description
-a	Sets device allocation class, ALLCLASS.
-i	Selective interactive mode, for setting all parameters.
-n	Sets device node name, NODENAME (up to 16 characters).
-o	Overrides warning messages.
-u	Sets device unit number, UNITNUM.

cdp

Option	Description
-sa *allclass*	Sets the allocation class (ALLCLASS) for all DSSI devices in the system to the value specified.
-sn	Sets the node name (NODENAME) for all DSSI devices in the system to either RFhscn or TFhscn, where h is the device hose number (0), s is the device slot number (0), c is the device channel number (0..3), and n is the device node ID number (0..6).
-su *unitnum*	Sets the starting unit number (UNITNUM) for the first DSSI device in the system to the value specified. The unit number for subsequent DSSI devices will be incremented from this base.

Examples

In the following example, the system lists the DSSI parameters for all DSSI drives on the system.

```
P00>>> cdp
    ❶            ❷            ❸            ❹ ❺  ❻
pua0.0.0.0.0    ALPHA0     0411214901371      2  0  $2$DIA0
pua0.1.0.0.0    ALPHA1     0411214901506      2  1  $2$DIA1
pua0.2.0.0.0    ALPHA2     041122A001625      2  2  $2$DIA2
pua0.3.0.0.0    ALPHA3     0411214901286      2  3  $2$DIA3
P00>>>
```

❶ Storage adapter device name

❷ Node name (NODENAME)

❸ System ID (SYSTEMID)

❹ Allocation class (ALLCLASS)

❺ Unit number (UNITNUM)

❻ Operating system device name

In the next example, the system sets the starting unit number for the first dua device in the system to the number 10. In increasing increments of 1, each subsequent dua device is also assigned a new unit number.

```
P00>>> cdp dua* -su 10

pua0.0.0.0.0    ALPHA0     0411214901371      2  10  $2$DIA10
pua0.1.0.0.0    ALPHA1     0411214901506      2  11  $2$DIA11
pua0.2.0.0.0    ALPHA2     041122A001625      2  12  $2$DIA12
pua0.3.0.0.0    ALPHA3     0411214901286      2  13  $2$DIA13
P00>>>
```

In the next example, the node name for all DSSI devices in the system is set to RFhscn using the default naming scheme (see the description of the -sn option).

```
P00>>> cdp -sn

pua0.0.0.0.0    RF0000     0411214901371      2  10  $2$DIA10
pua0.1.0.0.0    RF0001     0411214901506      2  11  $2$DIA11
pua0.2.0.0.0    RF0002     041122A001625      2  12  $2$DIA12
pua0.3.0.0.0    RF0003     0411214901286      2  13  $2$DIA13
P00>>>
```

In the next example, interactive editing mode is requested for the drive dua13, or RF0003 from the previous example. In the interaction that follows, the user sets the node name to ALPHA13, the allocation class to 1, and the unit number to 5.

```
P00>>> cdp -i dua13

pua13.3.0.0.0:
Node Name [RF0003]? ALPHA13
Allocation Class [2]? 1

Unit Number [13]? 5
P00>>>
```

continue

Synopsis

Resume program execution on the specified processor or on the primary processor if no processor is specified.

continue [*processor*]

Description

If you press the Halt button on the operator control panel and do not enter commands that disturb the system state, entering the `continue` command returns the system to operating system mode. If the computer is running the OpenVMS operating system, entering the `continue` command after halting the system with Ctrl/P has the same effect.

Arguments

Argument	Description
processor	The processor you specify to resume executing instructions.

Examples

In the following example, the system is commanded to resume operating system mode.

```
P00>>> continue
```

In the next example, the system's primary processor is commanded to resume operating system mode.

```
P00>>> continue &p0
```

ecu

Synopsis

Invoke the EISA Configuration Utility.

ecu

Description

Before you enter the `ecu` command, insert the ECU diskette for your operating system in the diskette drive.

Entering the `ecu` command from the `P00>>>` console prompt invokes the EISA Configuration Utility (ECU).

Before the system reads the ECU diskette, it first loads the ARC console firmware. Then it loads the ECU program.

After you complete the ECU procedure, you can switch back to the SRM console, if necessary, as described in the section ''Console Firmware Overview.''

For more information about running the EISA Configuration Utility, refer to your system Owner's Guide.

Example

In the following example, the EISA Configuration Utility is invoked from the SRM console.

```
P00>>> ecu
loading ARC firmware
.
.
.
```

edit

Synopsis

Invokes a console editor that behaves much like the line editor for the BASIC computer language. Used to add, insert, or delete lines in a RAM file or the ‘‘nvram’’ power-up script.

edit *file*

Subcommands

Subcommand	Function
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*.

Arguments

Argument	Description
file	The file to be edited.

Examples

The following example shows how to modify the user-defined power-up script, ‘‘nvram.’’ Explanatory comments are indicated by a pound sign (#). In this example, the script is edited so that the options in the system will be displayed each time the system is powered up.

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

                                Rev                    Events logged
 Slot   Option  Part#           Hw Sw   Serial#        SDD     TDD
   0    IO      B2222-AA        B2 0    NI53400099     00      00
   2    CPU0    B2024-AA        D3 11   ML43400022     00      00
   3    CPU1    B2024-AA        C3 11   ML43600045     00      00
   4    MEM0    B2021-CA        A1 0    GA33602859     00      00

 Slot   Option                  Hose 0, Bus 2, PCI on Primary I/O

 Slot   Option                  Hose 0, Bus 0, PCI on Primary I/O
   7    DECchip 21050-AA                               PCI Option Slot 5
   9    S3 Trio64/Trio32                               PCI Option Slot 7

 Slot   Option                  Hose 0, Bus 1, EISA on Primary I/O

P00>>> exit nvram
```

The following example shows how to clear the ''nvram'' script. To clear the script, enter line numbers without any text. This will delete the lines.

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

help or man

Synopsis

Display information about console commands.

help or man [*command . . .*]

Description

The `help` command, interchangeable with the `man` command, displays basic information about the use of console commands when the system is in console mode.

- If you enter the `help` command with no options or arguments, the system displays the complete list of commands.
- If you enter the `help` command followed by the name of a console command, the screen displays information about the console command.
- If you enter the `help` command with an argument string (such as ''sh''), the system displays information about all commands that begin with that string.

Arguments

Argument	Description
command . . .	Commands or topics for which help is requested.

Examples

In the following example, the system displays help on the `show` command.

```
P00>>> help show

NAME
         show
FUNCTION
         Display an environment variable value or other information.
SYNOPSIS
         show [{config,device,fru,memory,pal,version,...}]
                   [<envar>]
                   where:
                  <envar>={auto_action,bootdef_dev,boot_file,boot_osflags,...}

.
.
.
```

init

Synopsis

Initialize the console firmware.

init

Description

The `init` command performs a software reset only. No power-up diagnostics are performed. Use the Reset button on the operator control panel to initialize both the hardware and software.

Whenever you reset the following SRM environment variables, you must initialize the firmware with the `init` command to put the new setting into effect:

auto_action
bus_probe_algorithm
console
language
ocp_text
os_type
pk*0_fast
pk*0_host_id
pk*0_soft_term

Note

The `kbd_hardware_type` environment variable requires a hardware reset, using the Reset button.

Example

In the following example, the display device is switched from a graphics monitor to a serial terminal, and the firmware is initialized to put the new setting into effect.

```
P00>>> set console serial
P00>>> init
   .
   .
   .                 !Now switch to the serial terminal.
P00>>> show console
console             serial
```

man

See `help`.

more

Synopsis

Displays a file to the display device a screen at a time.

more [*file*. . .]

Description

This command is similar to the `more` command 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 is displayed, you can use the `more` command to display the contents of the event log file.

Arguments

Argument	Description
file	The name of the file to be displayed.

Examples

In the following example, the system is instructed to display the contents of the startup event log a screen at a time.

```
P00>>> more el
```

In the following example, the system is instructed to display the contents of online help a screen at a time.

```
P00>>> help * | more
```

set

Synopsis

Set or modify the value of an environment variable.

set [-default] *envar val*

Description

Sets or modifies the value of an environment variable. Environment 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.

Whenever you modify the value of the following environment variables, you must initialize the firmware with the `init` command.

auto_action
bus_probe_algorithm
console
kbd_hardware_type
language
ocp_text
os_type
pk*0_fast
pk*0_host_id
pk*0_soft_term

Note

The `kdb_hardware_type` environment variable requires a hardware reset, using the Reset button.

set

Arguments

Argument	Description
envar	The environment variable to be assigned a new value. Refer to the list of commonly used environment variables in Table 1–2.
val	The value that is assigned to the environment variable. Either a numeric value or an ASCII string.

Options

Option	Description
-default	Restores an environment variable to its default value.

Environment Variables

Table 1–2 provides a brief description of the commonly used environment variables. Each of these environment variables is described in the following sections. The environment variables for security commands are described in Chapter 2.

There are other internal environment variables used by the console for its own operation. These are normally not needed for system configuration and are not included here.

Note

Table 1–2 shows the default ROM settings. In some cases, configuration changes that override the ROM default are made at the factory. Use the `show` *`envar`* command to view the setting of an environment variable.

Table 1–2 Environment Variables

Variable	Function	ROM Default
auto_action	Specifies what action the console should take whenever the system is powered up, fails, or the Reset button is pressed.	Halt
bootdef_dev	Specifies the default boot device to the system.	System device on which Factory Installed Software (FIS) was loaded.
boot_osflags	Sets the operating system boot flags.	None
bus_probe_ algorithm	Sets bus probe algorithm to improve bus sizing and configuration on Digital UNIX and OpenVMS systems.	Digital UNIX V3.0B or later = new; OpenVMS 6.2 or later = new
com1_baud	Allows you to change the default baud rate of the COM1 serial port.	9600 baud
com2_baud	Allows you to change the default baud rate of the COM2 serial port.	9600 baud
console	Specifies the device on which power-up output is displayed.	Graphics
controlp	Halts the OpenVMS operating system.	On
cpu_enabled	Enables or disables specific CPUs on a symmetric multiprocessing system.	All CPUs are enabled.
ew*0_mode	Sets the default Ethernet device type on systems with a Digital Ethernet controller (ew*).	Depends on Ethernet device. AUI for most systems.
ew*0_protocols	Set network protocols for booting over the Ethernet controller (ew*).	MOP
kbd_hardware_ type	Sets the default console keyboard type.	PCXAL
language	Sets the console keyboard layout to a supported language.	English (American)
ocp_text	Specifies the message that is displayed on the power-up/diagnostic display.	Model *n/nnn*

(continued on next page)

set

Table 1–2 (Cont.) Environment Variables

Variable	Function	ROM Default
os_type	Specifies the operating system that will be run on the system. Sets the appropriate console user interface.	Matches the operating system installed.
pci_parity	Disables or enables parity checking on the PCI bus.	On (enabled)
pci_read_mult	Disables or enables a PCI bus read ahead.	Off
pk*0_fast	Enables Fast SCSI devices to perform in fast SCSI mode.	1 (Fast SCSI mode)
pk*0_host_id	Sets the default value for a controller host bus node ID.	Bus node ID 7
pk*0_soft_term	Enable or disable SCSI terminators on systems that use the QLogic ISP1020 SCSI controller.	Low
tga_sync_green	Set the location of the SYNC signal generated by the Digital ZLXp-E PCI graphics accelerator option.	ff (Should be reset to 00)
tt_allow_login	Enables or disables login to the SRM console firmware on alternative console ports.	1 (login enabled)

set auto_action

Synopsis

Set the console action at power-up and restart.

set auto_action *value*

Description

Sets the action the console should take any time the system is powered up, fails, or the Reset button is pressed. When the setting involves autobooting, the system boots from the default boot device that is specified by the value of the bootdef_dev environment variable.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

Value	Description
halt	Causes the system to remain in console mode after the system is powered up or it fails. ROM default.
boot	Causes the system to boot automatically when it is turned on. Causes the system to halt after a system failure.
restart	Causes the system to boot automatically when it is turned on or after it fails.

Example

In the following example, the system's default console action following error, halt, or power-up is changed from halt to boot.

```
P00>>> show auto_action
auto_action     halt
P00>>> set auto_action boot
P00>>> init
   .
   .
   .
P00>>> show auto_action
auto_action     boot
```

set bootdef_dev

Synopsis

Set the default boot device.

set bootdef_dev *boot_device*

Description

Specifies the default boot device to the system. The default boot device is the device from which the bootstrap system software is loaded. On Digital UNIX and OpenVMS systems, the default boot device has already been identified on your system as the device on which Factory Installed Software (FIS) was loaded.

Setting the default boot device simplifies the process of booting the operating system as follows:

- You can enter `b` and press Return at the console prompt and the system will find and boot operating system software.
- Assuming the default startup action is set to `boot` or `restart` (using the `set auto_action` command), the system will find and automatically boot the operating system at power-up.

Before setting the boot device, consider the following:

- Your boot device may already be set. (Enter the command `show bootdef_dev` to display the current default boot device.)
- You can set multiple boot devices. When you enter `b` or `boot` at the console prompt, the system searches for a bootable device from the list of devices that you specify. The system then automatically boots from the first device on which it finds software.
- You can override the default boot device by specifying an alternative device name on the `boot` command line.

Arguments

Argument	Description
boot_device	Name of a boot device. See the `show device` command for device naming conventions.

Example

In the following example, the default device from which the system attempts to boot is set to dka0.

```
P00>>> set bootdef_dev dka0
```

In the next example, the default devices are set to dub0 and dka0.

```
P00>>> set bootdef_dev dub0,dka0
P00>>> show bootdef_dev
bootdef_dev     dub0,dka0
```

set boot_osflags

Synopsis

Set the default boot flags and, in OpenVMS, a root number.

set boot_osflags *flags_value*

Description

Boot flags contain information used by the operating system to determine some aspects of a system bootstrap. Under normal circumstances, you can use the default boot flag settings.

To change the boot flags for the current boot only, you can use the *flags_value* option with the `boot` command.

Arguments

The argument *flags_value* is specific to the operating system.

Digital UNIX Systems

The Digital UNIX operating system takes a single ASCII character as an argument, as shown in Table 1–3.

Table 1–3 Digital UNIX Flag Values

Flags_Value	Meaning
a	Autoboot to multiuser mode.
i	Boot interactively to single user mode. Prompts for the name of a file to load and other options.
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 fails, it completes a partial memory dump. Specifying ‘‘D’’ forces a full dump.

OpenVMS Systems

The OpenVMS operating system expects an ordered pair: *root_number*, *boot_flags*.

root_number

The root number is the directory number on the system disk on which OpenVMS files are located. For instance:

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

boot_flags

Boot flag settings for OpenVMS systems are shown in Table 1–4.

You can specify multiple boot flags by adding the flag values together (logical ORing). For example, the flag value 10080 executes both the 80 and 10000 flag settings.

Table 1–4 OpenVMS Boot Flag Settings

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.

Examples

In the following Digital UNIX example, the boot flags are set to autoboot:

```
P00>>> set boot_osflags a
```

In the following OpenVMS example, *root_number* is set to 2 and *boot_flags* is set to 1. With this setting, the system will boot from root directory SYS2.SYSEXE to the SYSBOOT prompt.

```
P00>>> set boot_osflags 2,1
   .
   .
   .
```

In the following OpenVMS example, *root_number* is set to 0 and *boot_flags* is set to 80. With this setting, the operator will be prompted for the name of the secondary bootstrap file.

```
P00>>> set boot_osflags 0,80
   .
   .
   .
```

set bus_probe_algorithm

Synopsis

Set a bus probe algorithm on systems running Digital UNIX or OpenVMS.

set bus_probe_algorithm *mode_value*

Description

Setting this environment variable improves bus sizing and configuration on Digital UNIX systems and OpenVMS systems. It is not used for Windows NT systems. Set the bus probe algorithm for your system as indicated in the following table.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

Mode_Value	Description
Old	The bus probe algorithm should be set to old on OpenVMS V6.1 and earlier OpenVMS systems. Failure to do so could result in bugcheck errors when booting from an EISA device.
New	The bus probe algorithm should be set to new on Digital UNIX systems V3.0B or later and on OpenVMS systems V6.2 or later.

Example

In the following example, the bus probe algorithm is set to new.

```
P00>>> set bus_probe_algorithm new
P00>>> init
    .
    .
    .
P00>>> show bus_probe_algorithm
P00>>>
bus_probe_algorithm       new
```

set com1_baud or set com2_baud

Synopsis

Change the default baud rate of the COM1 or COM2 port.

set com1_baud *baud_value*, set com2_baud *baud_value*

Description

The default baud rate for the AlphaServer 2000, 2100, and 2100A systems is 9600. The `set com1_baud` and `set com2_baud` commands let you set the baud rate on the respective ports to other values.

Note

If you reset the baud rate, make sure that the new setting is compatible with the baud rate setting for the terminal.

Argument

The supported `baud_values` are 300, 600, 1200, 2400, 4800, 9600, and 19200.

Example

In the following example, the baud rate on COM1 is set to 19200.

```
P00>>> set com1_baud 19200
```

set console

Synopsis

Set the device on which power-up output is displayed.

set console *output_device*

Description

Power-up information is typically displayed on your console terminal. Your console terminal can be either a graphics monitor or a serial terminal. The setting of this environment variable determines where the system displays power-up output. Set this environment variable according to the console terminal that you are using.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

Output_Device	Description
graphics	Sets the power-up output to be displayed at a graphics monitor or a device that is connected to the VGA module at the rear of the system. ROM default.
serial	Sets the power-up output to be displayed on the device that is connected to the COM1 port at the rear of your system.

Example

In the following example, the display device is switched from a graphics monitor to a serial terminal, and the firmware is initialized to put the new setting into effect.

```
P00>>> set console serial
P00>>> init
   .
   .
   .                  !Now switch to the serial terminal.
P00>>> show console
console         serial
```

set controlp

Synopsis

Halt the OpenVMS operating system.

controlp [*value*]

Description

By default, the Ctrl/P key combination halts the OpenVMS operating system. If you need to use Ctrl/P as an application key, you can disable the default setting as shown in the example.

Arguments

Value	Description
On	The On setting enables Ctrl/P. ROM default.
Off	The Off setting disables Ctrl/P.

Example

In the following example, the Ctrl/P key combination is disabled, and entering Ctrl/P will no longer halt the OpenVMS operating system.

```
P00>>> set controlp off
P00>>> show controlp
controlp              off
```

set cpu_enabled

Synopsis

Enable or disable a CPU.

set cpu_enabled *hex_digit*

Description

This command 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; bit 1 corresponds to CPU 1, and so on. This is shown in Table 1–5 as the binary equivalent, 3210 (bit). A zero (0) bit prevents the corresponding CPU from running; a one (1) bit allows the corresponding CPU to run.

By default, the bit mask is all ones (*hex_digit* value F) so that CPU 0, CPU 1, CPU 2, and CPU 3 are all enabled.

To enable and disable CPUs, type the `set cpu_enabled` command followed by the *hex_digit* value for the CPU or CPUs you want to enable or disable. For example, to disable only CPU 1, you would enter the *hex_digit* value ''D.'' This value corresponds to enabling CPU 0, CPU 2, and CPU 3 and disabling CPU 1 (bit mask is set to 1101).

Arguments

The argument *hex_digit* takes the values shown in Table 1–5.

Table 1–5 Hex Digit Values for CPUs

Hex_Digit Value	Binary Equivalent CPU enable 3210 (bit)	Enabled CPUs
0	0000	No CPUs (CPU 0 still comes up)
1	0001	CPU 0
2	0010	CPU 1
3	0011	CPU 0,1
4	0100	CPU 2
5	0101	CPU 0,2
6	0110	CPU 1,2
7	0111	CPU 0,1,2
8	1000	CPU 3
9	1001	CPU 0,3
A	1010	CPU 1,3
B	1011	CPU 0,1,3
C	1100	CPU 2,3
D	1101	CPU 0,2,3
E	1110	CPU 1,2,3
F	1111	CPU 0,1,2,3

Examples

In the following example, CPUs 0, 2, and 3 are enabled, and CPU 1 is disabled.

```
P00>>> set cpu_enabled  d
```

In the following example, CPU 0 and CPU 1 are enabled, and CPU 2 and CPU 3 are disabled.

```
P00>>> set cpu_enabled 3
```

In the following example, CPU 3 is enabled, and CPU 0, 1, and 2 are disabled.

```
P00>>> set cpu_enabled  8
```

set ew*0_mode

Synopsis

Set an Ethernet controller to the Ethernet device type you specify. This command applies to systems with a Digital Ethernet controller (ew*).

set ew*0_mode *value*

Description

This command sets an Ethernet controller to run an AUI, ThinWire, or twisted-pair Ethernet network. For most devices, the ROM default is AUI. For the Fast setting, the device defaults to Fast.

- If the Ethernet controller is connected to AUI, set the default Ethernet device type to ‘‘aui.’’
- If the Ethernet controller is connected to ThinWire (BNC), set the default Ethernet device type to ‘‘bnc.’’
- If the Ethernet controller is connected to 10BaseT, set the default Ethernet device type to ‘‘twisted.’’
- If the Ethernet controller is set to 100BaseT, set the default Ethernet device type to ‘‘fast.’’

To list the Ethernet controllers on your system, enter the `show device` command. The Ethernet controllers start with the letters ‘‘ew,’’ for example, ewa0. The third letter is the adapter ID for the specific Ethernet controller. When entering the `set ew*0_mode` command, replace the asterisk with the adapter ID letter.

Arguments

The *value* depends on the network type, as shown in the following table. After you set the *value*, you must initialize the system with the `init` command.

Value	Description
a[ui]	Sets the default Ethernet device to AUI.
b[nc]	Sets the default Ethernet device to ThinWire.
fast	Sets the default Ethernet device type to fast for 100BaseT.
fastf[d]	Sets the default Ethernet device type to fast full duplex for 100BaseT.
fu[ll]	Sets the default Ethernet device type to full duplex, twisted pair.
t[wisted-pair]	Sets the default Ethernet device type to 10BaseT (twisted-pair).

Example

In the following example, the default Ethernet device type for Ethernet controller ‘‘a’’ is set to a 10BaseT twisted-pair controller.

```
P00>>> set ewa0_mode t
P00>>> init
   .
   .
   .
P00>>> show ewa0_mode
ewa0_mode     twisted-pair
```

set ew*0_protocols

Synopsis

Enable network protocols for booting and other functions. This command applies to systems with a Digital Ethernet controller.

set ew*0_protocols *protocol_value*

Description

To list the network devices on your system, enter the `show device` command. The Ethernet controllers start with the letters ''ew,'' for example, ewa0. The third letter is the adapter ID for the specific Ethernet controller. When entering the `set ew*0_protocols` command, replace the asterisk (*) with the adapter ID letter.

Arguments

Protocol_Value	Description
mop	Sets the network protocol to mop (Maintenance Operations Protocol): the setting typically used for systems using the OpenVMS operating system. ROM default.
bootp	Sets the network protocol to bootp: the setting typically used for systems using the Digital UNIX operating system.
bootp,mop	When both qualifiers are listed, the system attempts to use the mop protocol first, regardless of which protocol is listed first. If not successful, it then attempts the bootp protocol.

Examples

In the following example, the network protocol is set to mop for the device ewa0.

```
P00>>> set ewa0_protocols mop
P00>>> show ewa0_protocols
ewa0_protocols              mop
```

In the following example, the network protocol is set to bootp for device ewa0.

```
P00>>> set ewa0_protocols bootp
P00>>> show ewa0_protocols
ewa0_protocols              bootp
```

In the following example, the mop protocol is set to mop or bootp for device ewb0.

```
P00>>> set ewb0_protocols mop,bootp
P00>>> show ewb0_protocols
ewb0_protocols              bootp,mop
```

set host

Synopsis

Connect the console program to the MSCP DUP server on a DSSI device.

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

Description

Connects the console program to another DUP server on a DSSI (MSCP) device. In the syntax, *device* is the name of the device to attach to the console program. Using the -dup option invokes the DSSI DUP server on the selected device. You can use the DUP protocol to examine and modify parameters of a DSSI device.

For more information, refer to the *DSSI VMScluster Installation and Troubleshooting* manual.

Arguments

Argument	Description
device	Specifies the device to attach to the console program.

Options

Option	Description
-dup	Specifies connection to an MSCP DUP server. The DUP service may be used to examine and modify parameters of a DSSI device.
-task *task_name*	Specifies which DUP service utility to invoke. Refer to the example that follows for a list of utilities. If you do not specify this qualifier, a directory of utilities is displayed.

Example

The following example shows how to connect to the MSCP DUP server on a device.

```
P00>>> show device du
      dud0.0.0.3.0               R2YQYA$DIA0                    RF72
P00>>> set host -dup dud0
      starting DIRECT on pud0.0.0.3.0 (R2YQYA)

      Copyright (C) 1995  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 V1.1  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>>>
```

set kbd_hardware_type

Synopsis

Set the keyboard hardware type as either PCXAL (default) or LK411 for use with the `set language 3C (Francais)` command.

set kbd_hardware_type *keyboard_type*

Description

Your terminal keyboard layout depends on the language variant of the keyboard that you ordered. For systems using the language variant 3C (Francais), the keyboard hardware type must be set in order for the system to interpret the terminal keyboard layout correctly.

Whenever you change the value of this environment variable, you must perform a hardware initialization, using the Reset button.

Arguments

Keyboard_Type	Description
PCXAL	Selects the default keyboard hardware type.
LK411	Selects the LK411 keyboard layout for use with language variant 3C (Francais).

Example

In the following example, the keyboard hardware type is set for use with the French keyboard.

```
P00>>> set kbd_hardware_type LK411
   .
   .
   .                              !Press the Reset button.

P00>>> show kbd_hardware_type
kbd_hardware_type       LK411
```

set language

Synopsis

Set the keyboard layout, which is language dependent.

set language *language_code*

Description

Your terminal keyboard layout depends on the language variant of the keyboard that you ordered. In order for the system to interpret the terminal keyboard layout correctly, the console language setting must match the language of the keyboard variant that you ordered.

The system ships from the factory with a default keyboard setting of 36 English (American). Enter `show language` at the console prompt to display the current keyboard setting. If the current setting does not match your keyboard, reset this variable to match your keyboard variant.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

The values for *language_code* are as follows:

Code	Language
0	none (cryptic)
30	Dansk (Danish)
32	Deutsch (German)
34	Deutsch (Schweiz) (Swiss)
36	English (American)
38	English (British/Irish)
3A	Espanol (Spanish)
3C	Francais (French)
3E	Francais (Canadian)
40	Francais (Suisse Romande)
42	Italiano (Italian)
44	Nederlands (Netherlands)
46	Norsk (Norwegian)
48	Portuguese (Portuguese)
4A	Suomi (Finnish)
4C	Svenska (Swedish)
4E	Belgisch-Nederlands (Dutch)

Example

In the following example, the system is set to be used with the Spanish variant of a terminal keyboard.

```
P00>>> set language 3A
P00>>> init
   .
   .
   .
```

set ocp_text

Synopsis

Specify a message to be displayed on the power-up/diagnostic display.

set ocp_text *message*

Description

The power-up/diagnostic display on the front of the system is used to display important diagnostic and power-up information. When self-tests and diagnostics are completed, the CPU variant and system model number are displayed—for example, `Model 5/250`. You can use the `set ocp_text` command to display a different message of your choice. For example, you might want to display the network node name you have defined for the system.

The power-up/diagnostic display can display up to 16 characters.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

Value	Description
message	*your_text*

Example

In the following example, the system is set to display ‘‘Node Alpha1’’ on the power-up/diagnostic display when the operating system is up and running. The text to be displayed must be entered in quotation marks.

```
P00>>> set ocp_text "Node Alpha1"
P00>>> init
   .
   .
   .
```

set os_type

Synopsis

Set the default operating system.

set os_type *os_type*

Description

The operating system is set at the factory to Digital UNIX, OpenVMS, or Windows NT, depending on the operating system software you purchased. The `set os_type` command is used to change the factory default setting to the operating system you specify. The os_type environment variable also sets the firmware environment that will be started whenever the system powers up, is halted or restarted, or the operating system is shut down.

- If *os_type* is set to OpenVMS, the system starts the SRM firmware, and OpenVMS is the default operating system.
- If *os_type* is set to Digital UNIX or OSF, the system starts the SRM firmware, and Digital UNIX is the default operating system.
- If *os_type* is set to NT, the system starts the ARC firmware, and Windows NT is the default operating system.

If you decide to change operating systems, from Digital UNIX or OpenVMS to Windows NT or from Windows NT to Digital UNIX or OpenVMS, you must reset the os_type environment variable in order to activate the appropriate firmware for the operating system. If you change to Windows NT, you must also use the appropriate EISA Configuration Utility diskette.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

set os_type

Arguments

Os_Type	Description
NT	Sets the default operating system to Microsoft Windows NT; starts ARC firmware.
Digital UNIX or OSF	Sets the default operating system to Digital UNIX; starts SRM firmware.
OpenVMS	Sets the default operating system to OpenVMS; starts SRM firmware.

Example

In the following example, the default operating system is reset from OpenVMS to Microsoft Windows NT. After the system is initialized, the Windows NT Boot menu will be displayed.

```
P00>>> set os_type nt
P00>>> init
   .
   .
   .
```

set pci_parity

Synopsis

Disable or enable parity checking on the PCI bus.

set pci_parity *value*

Description

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. The `set pci_parity` command allows you to turn off PCI parity checking so that false PCI parity errors do not cause system problems.

Note

If you disable PCI parity, no parity checking is implemented by the host PCI bridge, so even those devices that generate good parity will be ignored.

Arguments

Value	Description
On	Enables PCI parity. ROM default.
Off	Disables PCI parity.

Example

In the following example, PCI parity is disabled.

```
P00>>> set pci_parity off
P00>>> show pci_parity
pci_parity          off
```

set pci_read_mult

Synopsis

set pci_read_mult *value*

Description

This environment variable controls a PCI bus read ahead. The OFF setting may improve performance of I/O options. For example, most EISA options do not benefit from a PCI bus read ahead.

Arguments

Value	Description
On	Enables a PCI bus read ahead.
Off	Disables a PCI bus read ahead. ROM default.

Example

In the following example, a PCI bus read ahead is disabled.

```
P00>>> set pci_read_mult
P00>>> show pci_read_mult
pci_read_mult      Off
```

set pk*0_fast

Synopsis

Enable either standard or Fast SCSI mode.

set pk*0_fast *scsi_speed*

Description

Enables Fast SCSI devices on a SCSI controller to perform in standard or fast mode. Not all controllers support the Fast SCSI setting.

- If you have at least one Fast SCSI device in your system, set the default controller speed to Fast SCSI (1).

  Devices on a controller that connects to both standard and Fast SCSI devices will automatically perform at the appropriate rate for the device, either fast or standard mode.

- If you have no Fast SCSI devices, set the default controller speed to standard SCSI (0).

  If a controller is set to standard SCSI mode, both standard and Fast SCSI devices will perform in standard mode.

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

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Note

Not all controllers support the option to set Fast SCSI mode.

set pk*0_fast

Arguments

SCSI_Speed	Description
0	Sets the default speed for devices on the controller to standard SCSI mode.
1	Sets the default speed for devices on the controller to Fast SCSI mode. ROM default.

Example

In the following example, the default speed for devices connected to a SCSI controller with an adapter ID of ‘‘a’’ is set to Fast SCSI mode.

```
P00>>> set pka0_fast 1
P00>>> init
    .
    .
    .
P00>>> show pka0_fast
pka0_fast     1
```

set pk*0_host_id

Synopsis

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

set pk*0_host_id *scsi_node_id*

Description

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 `show device` command. SCSI devices begin with the letters ‘‘pk,’’ for example pka0. The third letter is the adapter ID for the controller. When entering the `set pk*0_host_id` command, replace the asterisk with the adapter ID letter.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

SCSI_Node_ID	Description
0–7	Indicates the bus node ID.

Example

In the following example, the default bus node ID for a SCSI controller with an adapter ID of ‘‘b’’ is set to bus node ID 6.

```
P00>>> set pkb0_host_id 6
P00>>> init
    .
    .
    .
P00>>> show pkb0_host_id
pkb0_host_id     6
```

set pk*0_soft_term

Synopsis

Enable or disable SCSI terminators. This command applies to systems that use the QLogic ISP1020 SCSI controller.

set pk*0_soft_term *value*

Description

The QLogic ISP1020 SCSI controller implements the 16-bit wide SCSI bus. The QLogic module has two terminators, one for the low 8 bits and one for the high 8 bits.

Arguments

Value	Meaning
off	Turns off both low 8 bits and high 8 bits.
low	Turns on low 8 bits and turns off high 8 bits. ROM default.
high	Turns on high 8 bits and turns off low 8 bits.
on	Turns on both low 8 bits and high 8 bits.
diff	Places the bus in differential mode.

When entering this command, replace the asterisk (*) with the SCSI controller letter for which you are setting the default, for example, pkb0.

To get a list of the SCSI bus controllers on your system, enter the `show device` or `show config` command. A list of all system devices is displayed. The adapter ID is shown next to the QLogic ISP1020 listing.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Examples

In the following example, both terminators are disabled.

```
P00>>> set pkb0_soft_term off
P00>>> init
   .
   .
   .
P00>>> show pkb0_soft_term
pkb0_soft_term         off
```

In the following example, the terminator for the high 8 bits is enabled.

```
P00>>> set pkb0_soft_term high
P00>>> init
   .
   .
   .
P00>>> show pkb0_soft_term
pkb0_soft_term        high
```

set tga_sync_green

Synopsis

Set the location of the SYNC signal generated by the optional ZLXp-E PCI graphics accelerator (PBXGA).

set tga_sync_green *value*

Description

The ROM default for this environment variable is ff. The default must be reset to 00 so that the graphics monitor will synchronize. The correct setting, displayed with the `show` command is:

```
P00>>> show tga_sync_green
tga_sync_green          0
```

The parameter is a bit mask, where the least significant bit (lsb) sets the vertical SYNC for the first graphics card found, the second for the second found, and so on.

Arguments

Value	Description
ff	Synchronizes the graphics monitor on systems that do not use a ZLXp-E PCI graphics accelerator. ROM default.
00	Synchronizes the graphics monitor on systems with a ZLXp-E PCI graphics accelerator.

Example

The following command sets all graphics cards to synchronize on a separate vertical SYNC line, as required by some monitors. See your monitor documentation for all other information.

```
P00>>> set tga_sync_green 00
P00>>> show tga_sync_green
tga_sync_green              0
```

set tt_allow_login

Synopsis

Enable or disable login to the SRM console firmware on alternative console ports.

set tt_allow_login *value*

Description

If the console output device is set to ''serial,'' `set tt_allow_login 1` allows you to log in on the primary COM1 port, or alternate COM2 port, or the graphics monitor.

If the console output device is set to ''graphics,'' `set tt_allow_login 1` allows you to log in through either the primary graphics monitor or the alternate COM1 or COM2 console ports.

Whenever you change the value of this environment variable, you must initialize the firmware with the `init` command.

Arguments

Value	Description
0	Disables login on alternative console ports.
1	Enables login on alternative console ports. ROM default.

Example

In the following example, the primary console device is set to the terminal connected through the COM1 port. Then the `set tt_allow_login` command is used to disable logins through either the COM2 port or a graphics monitor.

```
P00>>> set console serial
P00>>> set tt_allow_login 1
P00>>> init
   .
   .
   .
```

show

Synopsis

Display an environment variable value or other information.

show [*envar*] [{config,device,fru,memory,pal,version}]

Description

This command displays the current value (or setting) for an environment variable that you specify. The SRM environment variables are listed in Table 1–2, earlier in this chapter.

Alternatively, this command displays other information about the system, according to the parameters that you enter on the command line. For example, you can display the system configuration by entering `show config`.

Arguments

Parameter	Description
envar	Displays the value of the environment variable specified.
config	Displays the current system configuration.
device	Displays devices and controllers in the system.
fru	Displays the serial number and revision level of system bus options and any errors for those options.
memory	Displays the memory module configuration.
pal	Displays the Privileged Architecture Library code (PALcode) version.
version	Displays the console program version.

Examples

In the following example, the system displays the version of the console program installed. The console program version is V4.3-140.

```
P00>>> show version
version                 V4.3-140  Nov  7 1995 13:59:28
P00>>>
```

In the next example, the default system power-up action is displayed.

```
P00>>> show auto_action
auto_action      boot
P00>>>
```

In the next example, a system's default boot device is displayed. The default boot device in the example is ewa0.

```
P00>>> show bootdef_dev
bootdef_dev      ewa0.0.0.3000.0
P00>>>
```

In the next example, a system's memory is displayed.

```
P00>>> show memory

Module   Size    Base Addr   Intlv Mode  Intlv Unit  Status
------   -----   ---------   ----------  ----------  ------
  0       64MB   00000000      1-Way         0       Passed
  1       32MB   04000000      1-Way         0       Passed
Total Bad Pages 0
P00>>>
```

show config

Synopsis

Display the system logical configuration.

show config

Description

This command displays logical devices found on the system bus, PCI bus, and EISA bus. You can use the information in the display to identify target devices for commands such as `boot` and `test`, as well as to verify that the system sees all the devices that are installed.

The configuration display varies somewhat for the AlphaServer 2000, 2100, and 2100A systems. The following examples illustrate each system.

AlphaServer 2100A Example

```
P00>>> show config
                      Digital Equipment Corporation
                         AlphaServer 2100A 5/300

SRM Console V4.3-134            VMS PALcode V1.15-3, OSF PALcode V1.18-2

Component       Status          Module ID
CPU 0              P            B2040-BA DECchip (tm) 21164-1
CPU 1              P            B2040-BA DECchip (tm) 21164-1
Memory 0           P            B2021-CA 128 MB
I/O                             B2222-AA
                                dva0.0.0.1000.0         RX26

 Slot   Option                  Hose 0, Bus 0, PCI
   2    Intel 82375EB                                   Bridge to Bus 1, EISA
   3    DECchip 21050-AA                                Bridge to Bus 2, PCI
   7    DECchip 21050-AA                                Bridge to Bus 3, PCI
   9    S3 Trio64/Trio32

 Slot   Option                  Hose 0, Bus 1, EISA

 Slot   Option                  Hose 0, Bus 2, PCI

   1    NCR 53C810              pka0.7.0.2001.0         SCSI Bus ID 7
                                dka500.5.0.2001.0       RRD45

 Slot   Option                  Hose 0, Bus 3, PCI
   0    DECchip 21040-AA        ewa0.0.0.3000.0         08-00-2B-E4-AE-A4
   1    QLogic ISP1020          pkb0.7.0.3001.0         SCSI Bus ID 7
                                dkb0.0.0.3001.0         RZ28
                                dkb100.1.0.3001.0       RZ28
                                dkb200.2.0.3001.0       RZ26L

P00>>>
```

- **Component, Status, and Module ID**

  These columns display the core system, including CPUs, memory, and I/O, and indicate their power-up status: P (pass) or F (fail).

- **Hose 0, Bus 0, PCI**
  - All controllers on Hose 0, Bus 0 of the primary PCI bus. The logical slot numbers are listed in the left column of the example.
  - Slot 0–1 = Reserved
  - Slot 2 = PCI to EISA bridge chip
  - Slot 3 = PCI to PCI bridge chip

- Slots 4–5 = Reserved
- Slots 6–9 = Correspond to physical PCI card cage slots on the primary PCI bus:

  Slot 6 = PCI 4
  Slot 7 = PCI 5
  Slot 8 = PCI 6
  Slot 9 = PCI 7
  In the case of storage controllers, the devices off the controller are also displayed.

- **Hose 1, Bus 0, EISA**

  All controllers on Hose 1, Bus 0, of the EISA bus. The logical slot numbers in the left column of the example correspond to the EISA card cage slots:

  Slot 1 = EISA 1
  Slot 2 = EISA 2
  Slot 3 = EISA 3

  In the case of storage controllers, the devices off the controller are also displayed.

- **Hose 0, Bus 2, PCI**
  - All controllers on Hose 0, Bus 2, of the secondary PCI bus. The logical slot numbers are listed in the left column of the example.
  - Slot 0 and 2–5 = Reserved
  - Slot 1 = SCSI controller on system backplane, along with storage drives on the bus.
  - Slot 6–9 = Correspond to physical PCI card cage slots on the secondary PCI bus:

    Slot 6 = PCI 0
    Slot 7 = PCI 1
    Slot 8 = PCI 2
    Slot 9 = PCI 3

AlphaServer 2100 Example

```
P00>>> show config

                        Digital Equipment Corporation
                           AlphaServer 2100 5/250


SRM Console V4.3-144            VMS PALcode V1.15-4, OSF PALcode V1.18-3

Component       Status          Module ID
CPU 0              P            B2040-AA DECchip (tm) 21164-1
Memory 0           P            B2021-BA  64 MB
Memory 1           P            B2021-AA  32 MB
I/O                             B2110-AA
                                dva0.0.0.1000.0         RX26

 Slot   Option                  Hose 0, Bus 0, PCI
   0    DECchip 21040-AA        ewa0.0.0.0.0            08-00-2B-E2-50-2C
   1    NCR 53C810              pka0.7.0.1.0            SCSI Bus ID 7
                                dka0.0.0.1.0            RZ28
                                dka200.2.0.1.0          RZ25L
                                dka600.6.0.1.0          RRD45
                                mka500.5.0.1.0          TLZ06
   2    Intel 82375EB                                   Bridge to Bus 1, EISA
   6    DEC KZPSA               pkb0.7.0.6.0            SCSI Bus ID 7
                                dkb200.2.0.6.0          RZ26L
                                dkb300.3.0.6.0          RZ28
   7    Mylex DAC960            dra.0.0.7.0
                                dra0.0.0.7.0            1 Member JBOD
                                dra1.0.0.7.0            1 Member JBOD
                                dra2.0.0.7.0            1 Member JBOD
                                dra3.0.0.7.0            1 Member JBOD
   8    DECchip 21040-AA        ewc0.0.0.8.0            08-00-2B-E2-1E-43

 Slot   Option                  Hose 0, Bus 1, EISA
   2    DE422
   3    DE425                   ewb0.0.0.1003.0         08-00-2B-94-F0-32
   5    DEFEA                   fra0.0.0.1005.0         00-03-08-00-00-03
   8    Compaq Qvision

P00>>>
```

- **Component, Status, and Module ID**

  CPU, memory, and standard I/O are shown with the results of power-up tests: P (pass) or F (fail).

- **Hose 0, Bus 0, PCI**
  - Slot 0 = Ethernet adapter (ewa0)
  - Slot 1 = SCSI controller on standard I/O, along with storage drives on the bus.
  - Slot 2 = EISA to PCI bridge chip
  - Slots 3–5 = Reserved
  - Slots 6–8 = Correspond to physical PCI card cage slots.

    Slot 6 = PCI 0
    Slot 7 = PCI 1
    Slot 8 = PCI 2

  - In the case of storage controllers, the devices off the controller are also displayed.

- **Hose 1, Bus 0, EISA**

  Slot numbers correspond to EISA card cage slots.

  Slot 1 = EISA 1
  Slot 2 = EISA 2
  Slot 3 = EISA 3
  Slot 4 = EISA 4
  Slot 5 = EISA 5
  Slot 6 = EISA 6
  Slot 7 = EISA 7
  Slot 8 = EISA 8

  In the case of storage controllers, the devices off the controller are also displayed.

- **Hose 2, Bus 0, PCI**

  Reserved

AlphaServer 2000 Example

```
P00>>> show config
                        Digital Equipment Corporation
                            AlphaServer 2000 4/233

SRM Console V4.3-140            VMS PALcode X5.48-117, OSF PALcode X1.35-81

Component       Status          Module ID
CPU 0              P            B2020-BA DECchip (tm) 21064A-2
Memory 0           P            B2023-BA 128 MB
I/O                             B2111-AA
                                dva0.0.0.1000.0         RX26

 Slot   Option                  Hose 0, Bus 0, PCI
   1    NCR 53C810              pka0.7.0.1.0            SCSI Bus ID 7
                                dka0.0.0.1.0            RZ28
                                dka100.1.0.1.0          RZ28B
                                dka600.6.0.1.0          RRD45
   2    Intel 82375EB                                   Bridge to Bus 1, EISA
   8    DECchip 21050-AA                                Bridge to Bus 2, PCI

 Slot   Option                  Hose 0, Bus 1, EISA
   3    DE425                   ewa0.0.0.1003.0         08-00-2B-94-D4-BA

 Slot   Option                  Hose 0, Bus 2, PCI
   0    DECchip 21040-AA        ewb0.0.0.2000.0         08-00-2B-E2-70-57
   1    QLogic ISP1020          pkb0.7.0.2001.0         SCSI Bus ID 7

P00>>>
```

- **Component, Status, and Module ID**

  CPU, memory, and I/O are shown with the results of power-up tests: P (pass) or F (fail).

- **Hose 0, PCI**
  - Slot 1 = SCSI controller on I/O backplane, along with storage drives on the bus.
  - Slot 2 = EISA to PCI bridge chip
  - Slots 3–5 = Reserved
  - Slots 6–8 = Correspond to physical PCI card cage slots:

    Slot 6 = PCI 0
    Slot 7 = PCI 1
    Slot 8 = PCI 2

  - In the case of storage controllers, the devices off the controller are also displayed.

- **Hose 1, EISA**

  Slot numbers 1–7 correspond to EISA card cage slots.

  Slot 1 = EISA 1
  Slot 2 = EISA 2
  Slot 3 = EISA 3
  Slot 4 = EISA 4
  Slot 5 = EISA 5
  Slot 6 = EISA 6
  Slot 7 = EISA 7
  Slot 8 = EISA 8

  In the case of storage controllers, the devices off the controller are also displayed.

- **Hose 2, PCI**

  Reserved

show device

Synopsis

Display system devices and controllers.

show device [*device_name*]

Arguments

Argument	Description
device_name	The device name or device abbreviation. When abbreviations or wildcards are used, all devices that match the type are displayed.

Device Naming

The device name conventions for the devices and controllers listed with the `show device` command are shown in Figure 1–2 and explained in the corresponding tables. Note that the logical I/O slot numbering in AlphaServer 2100A systems is different from that in AlphaServer 2000 and 2100 systems.

Figure 1–2 Device Naming Convention

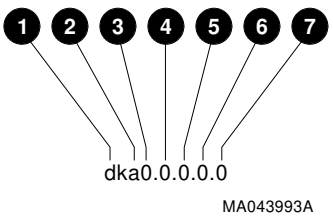


Table 1–6 Device Naming Convention: AlphaServer 2100A

Category	Description
❶ Driver ID	Two-letter port or class driver designator DR–RAID set device DV–Diskette drive EW–Ethernet port (PCI), FW–FDDI disk PK–SCSI port, DK–SCSI disk, MK–SCSI tape PU–DSSI port, DU–DSSI disk, MU–DSSI tape
❷ Storage Adapter ID	One-letter storage adapter designator (A,B,C . . .)
❸ Device Unit Number	Unique device unit number (MSCP unit number). SCSI unit numbers are forced to 100 X node ID.
❹ Bus Node Number	Bus node ID
❺ Channel Number	Used for multi-channel devices.
❻ Logical Slot Number	For AlphaServer 2100A: For EISA options—Correspond to EISA card cage slot numbers (1–3). For PCI options: Slot 0 = Reserved Slot 1 = SCSI controller on system backplane Slot 2 = PCI to EISA bridge chip Slot 3 = PCI to PCI bridge chip Slots 4–5 = Reserved Slots 6–9 = (primary bus) Correspond to physical PCI card cage slots: PCI 4, PCI 5, PCI 6, and PCI 7. Slots 6–9 (secondary bus). Correspond to physical PCI card cage slots: PCI 0, PCI 1, PCI 2, and PCI 3.
❼ Hose Number	For AlphaServer 2100A 0 PCI_0 (32-bit PCI); 1 EISA

Table 1–7 Device Naming Convention: AlphaServer 2000 and 2100

Category	Description
❶ Driver ID	Two-letter port or class driver designator DR–RAID set device DV–Diskette drive EW–Ethernet port (PCI), FW–FDDI disk PK–SCSI port, DK–SCSI disk, MK–SCSI tape PU–DSSI port, DU–DSSI disk, MU–DSSI tape
❷ Storage Adapter ID	One-letter storage adapter designator (A,B,C . . .)
❸ Device Unit Number	Unique device unit number (MSCP unit number). SCSI unit numbers are forced to 100 X node ID.
❹ Bus Node Number	Bus node ID
❺ Channel Number	Used for multi-channel devices.
❻ Logical Slot Number	For AlphaServer 2000 and 2100: For EISA options—Correspond to EISA card cage slot numbers (1–*). For PCI options: Slot 0 = Reserved Slot 1 = SCSI controller on AlphaServer 2100 system backplane or AlphaServer 2000 I/O backplane Slot 2 = PCI to EISA bridge chip Slots 3–5 = Reserved Slots 6–8 = Correspond to physical PCI card cage slots: PCI 0, PCI 1, PCI 2
❼ Hose Number	For AlphaServer 2000 and 2100: 0 PCI_0 (32-bit PCI) 1 EISA 2 PCI_1

AlphaServer 2100A Example

```
P00>>> show device
    ❶                    ❷       ❸                  ❹   ❺

dka500.5.0.2001.0        DKA500                     RRD45  1084
dkb0.0.0.3001.0          DKB0                        RZ28  442E
dkb100.1.0.3001.0        DKB100                      RZ28  442E
dkb200.2.0.3001.0        DKB200                     RZ26L  440C
dva0.0.0.1000.0          DVA0                        RX26
ewa0.0.0.3000.0          EWA0             08-00-2B-E4-AE-A4
pka0.7.0.2001.0          PKA0                 SCSI Bus ID 7
pkb0.7.0.3001.0          PKB0                 SCSI Bus ID 7  2.10
P00>>>
```

❶ Full console device name

❷ Abbreviated device name

❸ Node name. An alphanumeric node name of up to 6 characters can be set (not shown).

❹ Device type

❺ Firmware version (if known)

AlphaServer 2100 Example

```
P00>>> show device
    ❶                    ❷       ❸                  ❹   ❺
dka0.0.0.1.0             DKA0                        RZ28  T436
dka200.2.0.1.0           DKA200                     RZ25L  0006
dka600.6.0.1.0           DKA600                     RRD45  1084
dkb200.2.0.6.0           DKB200                     RZ26L  X442
dkb300.3.0.6.0           DKB300                      RZ28  435E
dra0.0.0.7.0             DRA0                1 Member JBOD
dra1.0.0.7.0             DRA1                1 Member JBOD
dra2.0.0.7.0             DRA2                1 Member JBOD
dra3.0.0.7.0             DRA3                1 Member JBOD
dva0.0.0.1000.0          DVA0                        RX26
mka500.5.0.1.0           MKA500                     TLZ06  0435
ewa0.0.0.0.0             EWA0            08-00-2B-E2-50-2C
ewb0.0.0.1003.0          EWB0            08-00-2B-94-F0-32
ewc0.0.0.8.0             EWC0            08-00-2B-E2-1E-43
pka0.7.0.1.0             PKA0                SCSI Bus ID 7
pkb0.7.0.6.0             PKB0                SCSI Bus ID 7   E01  A09
P00>>>
```

❶ Full console device name

❷ Abbreviated device name

❸ Node name. An alphanumeric node name of up to 6 characters can be set (not shown).

❹ Device type

❺ Firmware version (if known)

AlphaServer 2000 Example

```
P00>>> show device
    ❶                    ❷       ❸                  ❹ ❺

dka0.0.0.1.0             DKA0                        RZ28  442E
dka100.1.0.1.0           DKA100                      RZ28B  0006
dka600.6.0.1.0           DKA600                      RRD45  1084
dva0.0.0.1000.0          DVA0                        RX26
ewa0.0.0.1003.0          EWA0            08-00-2B-94-D4-BA
ewb0.0.0.2000.0          EWB0            08-00-2B-E2-70-57
pka0.7.0.1.0             PKA0                SCSI Bus ID 7
pkb0.7.0.2001.0          PKB0                SCSI Bus ID 7  2.10
P00>>>
```

❶ Console device name

❷ Abbreviated device name

❸ Node name. An alphanumeric node name of up to 6 characters can be set (not shown).

❹ Device type

❺ Firmware version (if known)

show fru

Synopsis

Display information about field-replaceable units (modules in the system).

show fru

Description

Displays module and error information for the following modules, based on the serial control bus EEPROM data:

- CPU modules
- Memory modules
- PCI and EISA modules

For each of the modules, the following items are displayed.

❶ System bus slot number for module (slots 1–7 top to bottom)

Slot 0: Dedicated EISA/PCI card cage slot on system backplane (AlphaServer 2100A and 2000) or on standard I/O module (AlphaServer 2100)
Slot 1–3, 5: CPU modules
Slot 4–7: Memory modules

❷ Option name (I/O, CPU#, or MEM#)

❸ Option part number

❹ Revision numbers (hardware and firmware)

❺ Option serial number

❻ Events logged:

Numbers other than ‘‘00’’ indicate that errors have been logged.

SDD: Number of symptom-directed diagnostic events logged by the serial ROM diagnostics at power-up.
TDD: Number of test-directed diagnostic events logged by the firmware diagnostics at power-up.

Examples

The following examples illustrate the output of the `show fru` command on the AlphaServer 2100A, 2100, and 2000 systems. The callouts in the examples are keyed to the description shown on this page.

AlphaServer 2100A Example

```
P00>>> show fru
  ❶    ❷      ❸                ❹        ❺              ❻
                                 Rev                    Events logged
 Slot   Option  Part#           Hw Sw   Serial#         SDD     TDD
   0    IO      B2222-AA        B2 0    NI53400096      00      00
   2    CPU0    B2020-AA        B2 10   KA426C0457      00      01
   4    MEM0    B2021-CA        A1 0    ML34398735      00      01

 Slot   Option                  Hose 0, Bus 2, PCI on Primary I/O

 Slot   Option                  Hose 0, Bus 0, PCI on Primary I/O
   6    DECchip 21050-AA                                PCI Option Slot 4
   9    Compaq 1280/P                                   PCI Option Slot 7

 Slot   Option                  Hose 0, Bus 1, EISA on Primary I/O
   1    DEC4220
P00>>>
```

AlphaServer 2100 Example

```
P00>>> show fru
  ❶    ❷      ❸                ❹        ❺              ❻
                                 Rev                    Events logged
   0    IO      B2110-AA        H2 0    KA427P0593      00      00
   2    CPU0    B2020-AA        B2 9    KA426C0457      00      00
   4    MEM0    B2021-BA        A1 0    ML34156292      00      00

 Slot   Option                  Hose 0, PCI

 Slot   Option                  Hose 1, EISA
   2    CPQ3011
   4    DEC4220

 Slot   Option                  Hose 2, PCI
P00>>>
```

show fru

AlphaServer 2000 Example

```
P00>>> show fru
  ❶    ❷      ❸              ❹        ❺              ❻
                               Rev                    Events logged
 Slot   Option  Part#          Hw Sw   Serial#         SDD     TDD
   0    IO      B2111-AA       A1 0    ML41100003      00      00
   2    CPU0    B2020-AA       B1 9    ML43400028      00      00
   5    MEM1    B2023-BA       A1 0    AY34915430      00      00

 Slot   Option                 Hose 0, PCI

   6    DECchip 21050-AA
   7    DECchip 21040-AA
   8    DECchip 21050-AA

 Slot   Option                 Hose 1, EISA
   1    ADP0001
   2    DEC4220
   5    DEC3002
   6    DEC4250
   7    CPQ3011

 Slot   Option                 Hose 2, PCI
P00>>>
```

show memory

Synopsis

Display information for each memory module in the system.

show memory

Description

This command displays information about each memory module, as indicated in the callouts.

Example

```
P00>>> show memory

  ❶       ❷       ❸          ❹          ❺         ❻
Module   Size    Base Addr   Intlv Mode  Intlv Unit  Status
------   -----   ---------   ----------  ----------  ------
  0       64MB   00000000      1-Way         0       Passed
Total Bad Pages 0 ❼
P00>>>
```

❶ Module slot number

❷ Size of memory module

❸ Base or starting address of memory module

❹ Interleave mode—number of modules interleaved (1–4-way interleaving)

❺ Interleave unit number

❻ Status (passed, failed, or not configured)

❼ Number of bad pages in memory (8 KB/page)

show pal

Synopsis

Display the PALcode version.

show pal

Description

This command displays the PALcode version. PALcode is the Alpha Privileged Architecture Library code, written to support Alpha processors. It implements architecturally defined behavior.

Example

In the following example, the system displays the versions of the PALcode that are installed on the system. The Digital UNIX PALcode version is X1.35-81. The OpenVMS PALcode version is X5.48-117.

```
P00>>> show pal
pal      VMS PALcode X5.48-117, OSF PALcode X1.35-81
P00>>>
```

show version

Synopsis

Display the console program version.

show version

Description

This command displays the version of the console program that is installed on the system.

Example

In the following example, the system displays the version of the console program that is installed on the system. The console program version is V4.3-140.

```
P00>>> show version
version                 V4.3-140 Nov 7 1995 13:59:28
P00>>>
```

test

Synopsis

Test the system.

test

Description

This command runs firmware diagnostics for the entire core system. The tests are run sequentially, and the status of each subsystem test is displayed to the console terminal as the tests progress. If a particular device is not available to test, a message is displayed.

Note

If you are running the Microsoft Windows NT operating system, switch from the ARC to the SRM console firmware in order to enter the `test` command. Refer to ''Console Firmware Overview'' at the beginning of this chapter for information about switching from the ARC to the SRM console.

The `test` script tests devices in the following order:

1. Memory tests (1 pass)
2. Read-only tests: DK* disks, DR* disks, DU* disks, DV* diskettes, MK* tapes, MU* tapes
3. Console loopback tests if lb argument is specified: COM2 serial port and parallel port
4. VGA console tests—These tests are run only if the console environment variable is set to ''serial.'' The VGA console test displays rows of the word ''digital.''
5. Network external loopback tests for EWA0—This test requires that the Ethernet port be terminated or connected to a live network or the test will fail.

Note

By default, no write tests are performed on disk and tape drives. Media must be installed to test the diskette drive and tape drives.

Examples

In the following example, the system is tested, and the tests complete successfully.

```
P00>>> test

Testing the Memory
Testing the DK* Disks(read only)
dkb600.6.0.2.1 has no media present or is disabled via the RUN/STOP switch
file open failed for dkb600.6.0.2.1
No DR* Disks available for testing
Testing the MK* Tapes(read only)
Testing the DV* Floppy Disks(read only)
file open failed for dva0.0.0.0.1
Testing the VGA(Alphanumeric Mode only)
Testing the EW* Network
P00>>>
```

In the next example, the system is tested, and the system reports an error message. The example indicates that no network server responded to a loopback message. Ethernet connectivity on this system should be checked.

```
P00>>> test

Testing the Memory
Testing the DK* Disks(read only)
No DR* Disks available for testing
Testing the MK* Tapes(read only)
Testing the DV* Floppy Disks(read only)
Testing the VGA(Alphanumeric Mode only)
Testing the EW* Network

*** Error (ewa0), Mop loop message timed out from: 08-00-2b-3b-42-fd

*** List index: 7 received count: 0 expected count 2

P00>>>
```

2

SRM Console Firmware Security Features

Introduction

This chapter explains how to implement security features on the SRM console. The security features are intended to prevent unauthorized persons from tampering with the system from the SRM console interface.

In This Chapter

This chapter contains the following topics:

- Security Features Overview
- Security Commands
- Secure Mode Functions

Security Features Overview

The SRM console firmware V3.9 and later contains console security features intended to prevent unauthorized personnel from modifying the system parameters or otherwise tampering with the system from the console. The security features include new commands and new parameters to some existing commands. The following two console modes are supported:

- Secure mode allows you to perform a limited number of console commands: `start`, `continue`, `boot` (with stored parameters), and `login` commands.
- User mode allows you to perform all SRM console commands.

Note

The security features work only if access to the system hardware is denied to unauthorized personnel. Be sure to keep the front panel of the system locked and make the key available only to authorized personnel.

Security Commands

set password

The `set password` command allows you to set the console password for the first time or to change an existing password. You do not have to set the password if you do not wish to operate in secure mode.

- If the password has been set previously, the console prompts you for the new password and verification, then prompts you for the old password. The password is not changed if the validation password entered does not match the existing password in the NVRAM.
- If the password has not been set, the console prompts you for the new password and verification.

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

Examples:

Set Password for the First Time (Successful)

```
P00>>> set password
Please enter the password: < password is not echoed >
Please enter the password again: < validation is not echoed >
P00>>>
```

Change an Existing Password (Successful)

```
P00>>> set password
Please enter the password:
Please enter the password again:
Now enter the old password:
P00>>>
```

Password Too Short Error

```
P00>>> set password
Please enter the password: < input is not echoed >
Password length must be between 15 and 30 characters
P00>>>
```

Password Validation Error

```
P00>>> set password
Please enter the password: < password is not echoed >
Please enter the password again: < validation is not echoed >
Validation error
P00>>>
```

set secure

The `set secure` command allows you to enable the security features without requiring you to restart the console. If the password has been set, the console will be secured and only the `start, continue, boot` (using the stored parameters), and `login` commands can be performed. If the password has not been set, you are prompted to set the password. See the `set password` command for instructions.

Examples:

Set Secure When Password Has Been Set

```
P00>>> set secure
Console is secure
P00>>>
P00>>> show boot_file
Console is secure
```

Set Secure When Password Has Not Been Set

```
P00>>> set secure
Secure not set. Please set the password.
P00>>>
```

login

The `login` command allows you to turn off the security features and gain access to all the SRM console commands during a particular session. The system automatically returns to secure mode as soon as you enter one of the following commands: `boot, continue, start` or when you initialize the system.

The `login` command works as follows: When you enter `login`, you are prompted for the current system password. If no password has been set, you are prompted that there is no password in NVRAM. You do not have to set a password if you do not wish to.

- If the password has been set, the following prompt is displayed:

  ```
  Please enter the password:
  ```

- If the password you enter matches the password in NVRAM when the prompt is redisplayed, the console is no longer in the secure mode and all the console commands can be performed.
- If you have forgotten the current password, use the `login` command in conjunction with the Halt button to clear the password, as described below:

  1. Make sure the Halt button is in the ''out'' position (not lit).
  2. Enter the `login` command:

     ```
     P00>>> login
     ```

  3. When the `Enter Password:` prompt is displayed, press the Halt button (the button should light up), then press Return.
  4. Press the Halt button to the ''out'' position (not lit). The password is now cleared and the console cannot be put into secure mode unless you set a new password.

Note

If you enter `login` with the Halt button in (lit), the command will fail, even if you entered the correct password. This prevents an intruder from depressing the Halt button and locking the front panel. Also, if you leave the Halt button in (lit) after you clear the password, the system will not boot.

Examples:

Valid Login When Secure Feature Is Not Set

```
P00>>> login
Secure not set. Please set the password.
P00>>>
```

Valid Login When Secure Feature Is Set

```
P00>>> login
Please enter the password: < password is not echoed >
P00>>>
```

Invalid Login When Secure Feature Is Set

```
P00>>> login
Please enter the password: < password is not echoed >
Invalid password
P00>>>
```

clear password

The `clear password` command clears the environment variable, password, 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. To use `clear password`, you must know the current password.

To clear the password without knowing the current password, you must use the `login` command in conjunction with the Halt button, as described in the section on the `login` command.

Examples:

Successful Clear Password Command

In this example, the `clear password` command was entered with the valid password:

```
P00>>> clear password
Please enter the password: < enter password (not echoed) >
Password successfully cleared
P00>>>
```

Unsuccessful Clear Password Command

In this example, the `clear password` command was entered with an invalid password:

```
P00>>> clear password
Please enter the password:  < invalid password entered
(not echoed) >
Console is secure
P00>>>
```

Secure Mode Functions

Secure mode allows you to perform only the boot command (using the stored parameters) and the login, continue, and start commands.

boot

When the console is in the secure mode, the boot command will not accept command line parameters. The console will boot 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 boot command will accept parameters when the console is not in secure mode.

Boot Examples

The following examples show a successful boot and an unsuccessful boot from a secure system.

Successful Boot from Secure System

```
P00>>> boot
(boot eza0.0.6.0 -file myfile.sys -flags 0)

Trying MOP boot.
.
.
.
```

Unsuccessful Boot from Secure System

```
P00>>> set boot_file myfile.sys
Console is secure
P00>>>
P00>>> boot -file myfile_ev4p2
Console is secure - parameters are not allowed
P00>>> boot -halt
console is secure - parameters are not allowed
P00>>> boot ezb0
Console is secure - parameters are not allowed

P00>>>
```

start and continue

The start and continue commands are valid on a secure console. After the commands are executed, the console is secured if there is a valid password. This prevents an intruder from halting the system and having access to a console that is not secure.

Ctrl/x

Ctrl/x will not create a foreground shell during the power-up script.

Ctrl/x is disabled during the power-up sequence to prevent an unauthorized user from initiating a foreground shell that may not be secure.

3

Using the ARC Console Interface

In This Chapter

This chapter explains how to use the ARC menu interface to carry out operations such as booting the system software and performing configuration tasks. The menu interface will be familiar to users of windowing systems.

ARC Utilities

The ARC console firmware is used primarily to configure and boot the system, and it is also used to run ARC-compliant utilities. The ARC console is described in this chapter and Chapter 4.

Two ARC utilities commonly used for configuring system options are the EISA Configuration Utility (ECU) and the RAID Standalone Configuration Utility.

EISA Configuration Utility

The ECU is used for configuring EISA options and some ISA options. You run the ECU under the following circumstances:

- When you add or move EISA options and some ISA options.
- When you remove EISA options or some ISA options from a system running ARC firmware lower than Version 4.38.

The ECU is run from diskette. Two diskettes are supplied with your hardware system shipment, one for Digital UNIX and OpenVMS and one for Windows NT. To run the ECU, refer to the instructions in the system Owner's Guide.

RAID Standalone Configuration Utility

The RAID Standalone Configuration Utility is used on systems that have optional StorageWorks RAID Array Subsystem controllers installed. You run the RAID Standalone Configuration Utility to set up the disk drives and logical units. For instructions, refer to *StorageWorks RAID Array 200 Subsystems Controller Installation and Standalone Configuration Utility User's Guide*, included in your StorageWorks RAID kit.

Serial-Line Support for ECU and RAID Utility

You can run the ECU and the RAID Standalone Configuration Utility from either a VGA-compatible graphics monitor or from a serial terminal (VT320 or higher, or equivalent). To use a serial terminal with these utilities, set up your terminal as follows:

1. Invoke the terminal setup utility as described in the documentation for your serial terminal, and change settings as follows:
   - From the General menu, set the terminal mode to VTxxx mode, 8-bit controls.
   - From the Comm menu, set the character format to 8 bit, no parity, and set receive XOFF to 128 or greater.
2. Enter the following commands at the SRM console prompt to set the console terminal to receive input in serial mode.

```
P00>>> set console serial
P00>>> init
   .
   .
   .                        !Now switch to the serial terminal.
P00>>> show console
console      serial
```

Switching Between Interfaces

To complete some procedures, it is necessary to switch from one console interface to the other.

- Several commands, including the `test` command, are run under the SRM interface.
- The EISA Configuration Utility (ECU) and the RAID Standalone Configuration Utility are run under the ARC interface.

Switching from SRM to ARC

To switch from the SRM console to the ARC console, enter the `arc` command at the SRM console prompt `P00>>>`.

Switching from ARC to SRM

Switch from the ARC console to the SRM console as follows:

1. From the Boot menu, select `Supplementary menu`.
2. From the Supplementary menu, select `Set up the system . . . .`
3. From the Setup menu, select `Switch to OpenVMS or Digital UNIX console.`
4. Select your operating system console, then select the Setup menu and press Enter.
5. When the `Power-cycle the system to implement the change` message is displayed, press the Reset button.

   Once the console firmware is loaded and the system is initialized, the SRM console prompt, `P00>>>`, is displayed.

ARC Interface Introduction

Summary

The ARC user interface is menu based. It has three primary menus: the Boot menu, the Supplementary menu, and the Setup menu.

Boot Menu Functions

The Boot menu contains selections that allow you to perform the following tasks:

- Boot the default boot selection
- Boot an alternate boot selection
- Run a program
- Access the Supplementary menu

Supplementary Menu Functions

The Supplementary menu contains selections that allow you to perform the following tasks:

- Install new firmware
- Install Microsoft Windows NT from a compact disc
- Access the Setup menu
- Display the hardware configuration
- Return to the Boot menu

Setup Menu Functions

The Setup menu contains selections that allow you to perform the following tasks:

- Set the system time
- Set the default system partition
- Configure the system floppy drives and keyboard
- Manage the ARC boot selections (Chapter 4)
- Set the system to boot automatically
- Run the EISA Configuration Utility (see your system documentation).
- Edit the system environment variables
- Reset the system to factory defaults
- Set the default operating system and system firmware
- Return to the Supplementary menu, with or without saving Setup menu changes

Language Selection

Versions of the ARC console firmware, numbered V4.1 or higher, are available in six languages: English, French, German, Italian, Japanese, and Spanish. The default language is English.

Language Support Files

A diskette containing the language support files is provided with new systems. If you wish to install a language other than English, locate the diskette before you proceed. Once you install the language of your choice, that language will be displayed whenever you use the ARC firmware menus. You can switch to a different language at any time.

Updated language support files are provided with firmware updates. You can download these updates from the following online locations:

- On CompuServe, in the DEC4WNT forum in the Library Section [4], Hardware Support.
- ftp://ftp.digital.com/pub/Digital/Alpha/systems/
- http://www.service.digital.com/alpha/server/

Language Selection for Updated Firmware

If your ARC Console firmware version is numbered lower than V4.1, you must first update your firmware before you can utilize the multi-language feature. For information on updating your firmware, follow the instructions that came with your Windows NT kit or refer to the *Firmware Update Utility User Guide and Release Notes* shipped with firmware updates.

After you update the firmware, you can change the language selection, if desired, as follows:

1. Turn on power to the system. After a short time, the Language selection menu is displayed.
2. From the Language selection menu, select `Other language`.
3. Insert the language diskette or CD-ROM into the appropriate drive.
4. From the Language menu that is displayed, select the language of your choice.

5. When prompted to save the language files to hard disk, select `yes`.
6. At the Choose hard disk location for language files: menu, select `Default`.

Note

While it is possible to save the language files to another disk or FAT partition, it is highly recommended that you accept the default location. Do not save language files to a disk that will be removed.

This completes the language installation.

Changing to Another Language

Use the following procedure if you are using the default language (English) and wish to change to another language or if you have already selected a language and wish to change it.

1. Turn on the power to your system. After a short wait, the ARC Boot menu is displayed.
2. Select `Supplementary menu`.
3. From the Supplementary menu, select `Set up the system`.
4. From the Setup menu, select `Set system language`.
5. Insert the language diskette or CD-ROM into the appropriate drive. If the language files are already on your hard drive, skip this step.
6. From the Language Selection menu, select `Other language`.
7. From the Language Location menu, select `Floppy Disk`, `CD-ROM`, or `Hard Disk` as appropriate.
8. From the Language menu that is displayed, select the language of your choice.

9. At the Save language files to hard disk? menu, select `yes`.
10. At the Choose hard disk location for language files: menu, select `Default`.

Note

While it is possible to save the language files to another disk or FAT partition, it is highly recommended that you accept the default location. Do not save language files to a disk that will be removed.

11. This completes the language change.

Accessing the Boot Menu

Summary

This section describes how to access the ARC firmware Boot menu.

Access Procedure

Table 3–1 lists the steps to access the ARC firmware Boot menu from different system states.

Table 3–1 Accessing the Boot Menu

Step	Action	Result
1	Shut down the operating system, using the instructions in the operating system documentation.	The system enters the ARC firmware if Windows NT was running, or enters the SRM firmware is Digital UNIX or OpenVMS was running.
2	If the `P00>>>` SRM prompt is displayed, enter the `set os_type nt` command to select the ARC firmware.	This command causes the system to enter the ARC firmware when the system is initialized.
3	Enter the `init` command (software reset) or turn the system off, then turn it on again (hardware reset).	On a hardware reset, the system completes the power-up tests, checks the EISA configuration, then displays the Boot menu. On a software reset, no hardware testing is done, and the Boot menu is displayed.

Boot Menu Example

Example 3–1 shows a sample Boot menu.

Example 3–1 Boot Menu Example

```
Alpha Firmware Version n.nn
Copyright (c) 1993--1995 Microsoft Corporation
Copyright (c) 1993--1995  Digital Equipment Corporation

Boot menu:

   Boot Windows NT
   Boot an alternate operating system
   Run a program
   Supplementary menu...

Use the arrow keys to select, then press Enter.
```

Booting the Default ARC Boot Selection

Summary

This section describes how to boot the default boot selection.

Booting the Default Boot Selection

Table 3–2 lists the steps to boot the default boot selection on your system.

Table 3–2 Booting the Default Boot Selection

Step	Action	Result
1	Access the ARC firmware Boot menu.	The system displays the Boot menu.
2	Choose the first menu item and press Enter. The first menu item takes the form `Boot <boot selection name>`. Usually, the boot selection name is Windows NT. Alternatively, you can wait for the system to autoboot the default boot selection.	The system displays boot messages as it boots Windows NT or displays an error message if the boot selection is incorrectly defined.

If Errors Occur

Refer to Chapter 4 if the system displays an error message when you attempt to boot the default boot selection.

Example

Example 3–2 shows the ARC firmware Boot menu.

Example 3–2 Choosing the Boot Windows NT Menu Item

```
ARC Multiboot Alpha Version n.nn
Copyright (c) 1993--1995 Microsoft Corporation
Copyright (c) 1993--1995 Digital Equipment Corporation

Boot menu:
   Boot Windows NT
   Boot an alternate operating system
   Run a program
   Supplementary menu...

Use the arrow keys to select, then press Enter.
Seconds until auto-boot, select another option to override: 30
```

Booting an Alternate Operating System

Summary

This section describes how to boot a different operating system.

Booting an Alternate System

Table 3–3 lists the steps to boot an alternate operating system.

Table 3–3 Booting an Alternate Operating System

Step	Action	Result
1	Access the ARC firmware Boot menu.	The system displays the Boot menu.
2	Choose `Boot an alternate operating system` and press Enter.	The system displays a selection menu listing the system boot selections.
3	Choose the operating system that you want to boot and press Enter.	The system displays boot messages as it boots the chosen selection or displays an error message if the selection is incorrectly defined.

If Errors Occur

See Chapter 4 if the system displays an error message when you attempt to boot an alternate operating system.

Example 1

Example 3–3 shows the ARC firmware Boot menu.

Example 3–3 Booting an Alternate Operating System

```
Boot menu:

   Boot the default operating system
   Boot an alternate operating system
   Run a program
   Supplementary menu...

Use the arrow keys to select, then press Enter.
```

Example 2

Example 3–4 shows the boot selection menu.

Example 3–4 Choosing the Alternate Operating System

```
Boot menu:

   Boot Windows NT  (Default)
   Boot Digital UNIX
   Boot OpenVMS

Use the arrow keys to select, then press Enter.
```

Running a Program

Summary

This section describes how to run a program using the `Run a program` menu item.

Digital supplies specially compiled and linked programs run from the ARC firmware to perform various tasks. Typical programs include the EISA Configuration Utility (ECU) and the RAID Standalone Configuration Utility. These programs can be run from any file allocation table (FAT) or NT file system (NTFS) formatted device.

Running a Program

Table 3–4 lists the steps to run a program from the ARC firmware Boot menu.

Table 3–4 Running a Program

Step	Action	Result
1	Access the ARC firmware Boot menu.	The system displays the Boot menu.
2	Choose `Run a program` and press Enter.	The system displays a prompt asking you to enter the name of the program.
3	Enter the name of the program, giving the complete device, path, and program name.	The system searches the device and path that you specify for the program. If it finds the program, it executes it. When you exit the program, the system displays the Boot menu again.

Example

The following example shows how to start the RAID Standalone Configuration Utility from a diskette. After you enter the device and program name, the utility's Main menu will be displayed.

```
Run a program: A:swxcrmgr
```

Accessing the Supplementary Menu

Summary

This section describes how to access the Supplementary menu in the ARC firmware.

Accessing the Supplementary Menu

Table 3–5 lists the steps to access the Supplementary menu.

Table 3–5 Accessing the Supplementary Menu

Step	Action	Result
1	Access the ARC firmware Boot menu.	The system displays the Boot menu.
2	Choose `Supplementary menu...` and press Enter.	The system displays the Supplementary menu.

Supplementary Menu Example

Example 3–5 shows the Supplementary menu.

Example 3–5 Accessing the Supplementary Menu

```
ARC Multiboot Alpha Version n.nn
Copyright (c) 1993--1995  Microsoft Corporation
Copyright (c) 1993--1995  Digital Equipment Corporation

Supplementary menu:

   Install new firmware...
   Install Windows NT from CD-ROM
   Set up the system...
   Display hardware configuration
   Boot menu...

Use the arrow keys to select, then press Enter.
```

Accessing the Setup Menu

Summary

This section describes how to access the Setup menu.

Accessing the Setup Menu

Table 3–6 lists the steps to access the Setup menu.

Table 3–6 Accessing the Setup Menu

Step	Action	Result
1	Access the Supplementary menu.	The system displays the Supplementary menu.
2	Choose `Set up the system...` and press Enter.	The system displays the Setup Menu.

Identifying and Correcting Setup Errors

The system can detect setup errors that may prevent the system from booting. When you access the setup menu, an arrow symbol (->) indicates the menu item that you must choose to correct the error. The menu item is highlighted in yellow. There are two methods that you can use to correct these errors:

- Choose each menu item that indicates an error, starting from the top of the menu, and correct the errors.
- Choose `Reset system to factory defaults`.

  See the section ‘‘Resetting the System to Factory Defaults ,’’ later in this chapter, for more information on this menu item.

Caution

Choosing the reset option deletes all boot selections and environment variables and replaces them with factory default values. Use this option with caution.

Setup Menu Example

Example 3–6 shows the Setup menu.

Example 3–6 Setup Menu

```
Setup menu:

     Set system time
     Set default environment variables
   ->Set default configuration
     Set system language
     Manage boot selection menu
     Setup autoboot

     Run EISA configuration utility from floppy
     Edit environment variables
     Reset system to factory defaults

     Help
     Switch to OpenVMS or Digital UNIX console
     Supplementary menu, and do not save changes...
     Supplementary menu, and save changes...

Use the arrow keys to select, then press Enter.
```

Note

The system language option is available with Version 4.1 or higher of the firmware.

Displaying the Hardware Configuration

Summary

This section describes how to display the hardware configuration. There are several parts to the hardware display:

- The first screen displays processor information, the amount of memory installed, and the type of video card installed.
- The second screen displays the boot devices.
- The third and fourth screens display information about the adapters installed in the system's EISA and PCI slots, respectively.

Displaying Hardware Configuration

Table 3–7 lists the steps to display the hardware configuration.

Table 3–7 Displaying the Hardware Configuration

Step	Action	Result
1	If necessary, access the Supplementary menu.	The system displays the Supplementary menu.
2	Choose `Display hardware configuration` and press Enter. Press any key to continue to the next screen.	The system displays the hardware configuration screens.

Boot Device Names

Table 3–8 explains the device names listed on the second screen of the hardware configuration display.

Note

The hardware configuration display does not list tape drives or network devices.

Table 3–8 Device Names

Name	Description
multi(0)key(0)keyboard(0) multi(0)serial(0) multi(0)serial(1)	The multi( ) devices are located on the system module. These devices include the keyboard port and the serial line ports.
eisa(0)video(0)monitor(0) eisa(0)disk(0)fdisk(0)	The eisa( ) devices are provided by devices on the EISA bus. These devices include the monitor and the floppy drive.
scsi(0)disk(0)rdisk(0) scsi(0)cdrom(6)fdisk(0)	The scsi( ) devices are SCSI disk or CD-ROM devices. These examples represent installed SCSI devices. The disk drives are set to SCSI ID 0, and the CD-ROM drive is set to SCSI ID 6. The devices have logical unit numbers of 0.

Hardware Configuration Display

Example 3–7 shows a sample hardware configuration display.

Example 3–7 Sample Hardware Configuration Display

```
Alpha Processor and System Information:

  Processor ID       21164
  Processor Revision 2.2
  System Revision    0x0
  Processor Speed    250.00 MHz
  Physical Memory    512 MB


Extended Firmware Information:

   Version: 4.43 951101.1202

   NVRAM Environment Usage: 52%
      (540 of 1024 bytes)

Video Option detected:

  24 Plane Frame Buffer video card:  ZLXp-E2

 Press any key to continue...
                                        Monday, 11-13-1995  5:05:38 PM
Devices detected by the firmware:

   eisa(0)video(0)monitor(0)
   multi(0)key(0)keyboard(0)
   eisa(0)disk(0)fdisk(0)     (Removable)
   multi(0)serial(0)
   multi(0)serial(1)
   scsi(0)disk(1)rdisk(0)     (RAW)           DEC     RZ28     (C) DECD41C
   scsi(0)disk(2)rdisk(0)     (RAW)           DEC     RZ28     (C) DECD41C
   scsi(0)disk(3)rdisk(0)     (RAW)           DEC     RZ28B    (C) DEC0006
** scsi(0)tape(5)tape(0)                      DEC     TLZ06    (C) DEC0491
   scsi(0)cdrom(6)fdisk(0)    (Removable)     DEC     RRD45    (C) DEC 1084
   scsi(1)disk(8)rdisk(0)     (2 Partitions)  MYLEX DAC960
   scsi(1)disk(9)rdisk(0)     (RAW)           MYLEX DAC960
   scsi(1)disk(10)rdisk(0)    (RAW)           MYLEX DAC960

 ** --> Devices detected but not supported by the firmware.

 Press any key to continue...
                                        Monday, 11-13-1995  5:05:38 PM

PCI slot information:

    Bus  Device Function  Vendor Device Revision  Interrupt    Device
  Number Number  Number     ID     ID      ID      Vector       Type
  ------ ------ --------  ------ ------ --------  ---------  ----------
     0      0       0      1011      2     23        22      Ethernet
     0      1       0      1000      1      1        21      SCSI
     0      2       0      8086    482      3         0      EISA bridge
     0      7       0      1011      4      3        24      Video card
```

(continued on next page)

Example 3–7 (Cont.) Sample Hardware Configuration Display

```
 Press any key to continue...

        Monday, 11-13-1995  5:05:38 PM
EISA slot information:

  Slot    Device          Identifier
    0     Other           DEC2A01
    4     Display         PHI8041
    6     Unknown         ADP0001
    7     Disk            MLX0075
    0     Disk            FLOPPY

 Press any key to continue...
        Monday, 11-13-1995  5:05:38 PM

 Copyright (c) 1993-1995 Microsoft Corporation
 Copyright (c) 1993-1995 Digital Equipment Corporation
```

Setting the System Date and Time

Summary

This section describes how to set the system date and time from the ARC firmware.

Setting the Date and Time

Table 3–9 lists the steps to set the system date and time.

Table 3–9 Setting the System Date and Time

Step	Action	Result
1	Access the Setup menu.	The system displays the Setup menu.
2	Choose the `Set system time` menu item and press Enter.	The system displays a prompt that asks you to enter the date and indicates the date format that it expects (mm-dd-yy).
3	Enter the date in the required format and press Enter.	The system saves the date that you set in the nonvolatile random-access memory (NVRAM). It then displays a prompt that asks you to enter the time and indicates the time format that it expects (hh-mm-ss).
4	Enter the time in the required format and press Enter.	The system saves the date that you set in the NVRAM, returns to the Setup menu, and displays the new day, date, and time at the top of the screen.

Set System Time Example

Example 3–8 shows how to set the system date and time to 11:59 a.m., December 31, 1996. Entries are in 24-hour time. For example, to set the time to 11:59 p.m., you would enter 23:59:00.

Example 3–8 Set System Time Example

```
        Sunday, 3-15-1995   12:13:33 AM

Enter the new date (mm-dd-yy) :  12-31-96
Enter time (hh:mm:ss) :  11:59:00
```

Setting the Default System Partition

Summary

This section describes how to set the default system partition. The system partition is the partition that contains the operating system loader (OSLOADER.EXE for Windows NT).

Setting the System Partition

Table 3–10 lists the steps to set the default system partition.

Table 3–10 Setting the Default System Partition

Step	Action	Result
1	Access the Setup menu.	The system displays the Setup menu.
2	Choose `Set default environment variables` and press Enter.	The system displays a list of three device types: SCSI Hard Disk, Floppy Disk, and CD-ROM.
3	Select one of the device types and press Enter.	The system displays different prompts depending on the device that you selected.
4	Choose the storage device containing the system partition to use. • If you chose `SCSI Hard Disk`, specify the SCSI ID and the partition number and press Enter. • If you chose `Floppy Drive`, specify the floppy drive number and press Enter. • If you chose `CD-ROM`, specify the SCSI ID and press Enter.	The firmware holds the values you specify in memory. See the section entitled ‘‘Saving the System Setup or Quitting the Setup Menu ’’ for information on saving these values.

(continued on next page)

Table 3–10 (Cont.) Setting the Default System Partition

Step	Action	Result
5	When you have finished making changes in the Setup menu, save your system setup. For information, see the section entitled ''Saving the System Setup or Quitting the Setup Menu .''	Your setup information is saved.

Default System Partition Example

Example 3–9 shows an example of setting up the default system partition as partition 1 on a SCSI disk drive with SCSI ID 1.

Example 3–9 Setting the Default System Partition

```
Enter location of default system partition:
     Select media:
   SCSI Hard Disk
   Floppy Disk
   CD-ROM
     Enter SCSI bus number: 0
     Enter SCSI ID: 1
     Enter partition (must be FAT or NTFS) : 1
```

Note

Refer to the Windows NT Release Notes for your system for information on the recommended partition arrangement.

Effects of This Procedure

Table 3–11 describes how this procedure affects other system settings and the actions that you must take as a result of using the procedure.

Table 3–11 Effects of Setting the Default System Partition

Effect	Suggested Action
Sets the SYSTEMPARTITION environment variable value.	None.
Sets the FWSEARCHPATH environment variable to the same value as the SYSTEMPARTITION environment variable.	None.
Clears the floppy drive information in the default system configuration.	For information on setting the floppy drive information in the system configuration, see the section ‘‘Configuring the System Floppy Drives and Keyboard Type ,’’ later in this chapter.
Clears all boot selection environment variables except SYSTEMPARTITION.	For information on adding a boot selection before you try to boot Windows NT, see Chapter 4.
Clears the AUTOLOAD and COUNTDOWN environment variables.	For information on setting the AUTOLOAD and COUNTDOWN environment variables, see the section entitled ‘‘Setting the System to Boot Automatically .’’

Configuring the System Floppy Drives and Keyboard Type

Summary

This section describes how to configure the Microsoft Windows NT system floppy drives and keyboard type.

Configuring Drives and Keyboard

Table 3–12 lists the steps to configure the system floppy drives and keyboard type.

Table 3–12 Configuring the Floppy Drives and Keyboard

Step	Action	Result
1	Access the Setup menu.	The system displays the Setup menu.
2	Choose `Set default configuration` and press Enter.	The system displays a prompt asking you to choose the floppy drive capacity.
3	Choose the first floppy drive's capacity and press Enter.	The firmware holds a value indicating the floppy drive capacity in memory. The system displays a prompt asking you to specify whether there is a second floppy drive.
4	Choose whether your system contains a second floppy drive. • If the system does not contain a second floppy drive, choose No and press Enter. Then go to step 5. • If the system contains a second floppy drive, choose Yes and press Enter. Enter a value for the floppy drive capacity. Then go to step 5.	If the system does not contain a second floppy drive, you are prompted for the keyboard language. If the system contains a second floppy drive, you are prompted for the floppy drive capacity.

(continued on next page)

Table 3–12 (Cont.) Configuring the Floppy Drives and Keyboard

Step	Action	Result
5	Choose the language for the keyboard that you ordered.	The system prompts you to select the keyboard from a list. The system holds a value indicating the keyboard language in memory, then displays the Setup menu.
6	When you have finished making changes in the Setup menu, save your system setup. See the section entitled ‘‘Saving the System Setup or Quitting the Setup Menu .’’	The system saves the changes.

Floppy Drive Configuration Example

Example 3–10 shows an example of configuring a system with two floppy drives.

Example 3–10 Configuring the System Floppy Drives

```
Select floppy drive capacity:
   5.25" 1.2MB
   3.5"  1.44MB
   3.5"  2.88MB

Is there a second floppy:
   Yes
   No

Select floppy drive capacity:
   5.25" 1.2MB
   3.5"  1.44MB
   3.5"  2.88MB
```

(continued on next page)

Example 3–10 (Cont.) Configuring the System Floppy Drives

```
Select keyboard:
   U.S.              101-key keyboard
   Japanese          106-key keyboard
   French            102-key keyboard
   German            102-key keyboard
   Spanish           102-key keyboard
   Spanish variation
   Canadian French   102-key keyboard
   Swiss             102-key keyboard
   Italian           102-key keyboard
   Finnish/Swedish keyboard
   Norwegian keyboard
   Danish            102-key keyboard
```

Effects of This Procedure

Table 3–13 describes how this procedure affects other system settings and the actions that you must take as a result of using the procedure.

Table 3–13 Effects of Configuring the System Floppy Drives

Effect	Suggested Action
Sets the FLOPPY environment variable to 0, 1, or 2.	None.
Sets the FLOPPY2 environment variable to N, 0, 1, or 2.	None.
Affects the EISA configuration.	Run the EISA Configuration Utility following the instructions listed in your system documentation.

Setting the System to Boot Automatically

Summary

This section describes how to set the system to boot automatically when you turn it on or when you switch from the SRM firmware to the ARC firmware.

Setting AutoBoot

Table 3–14 lists the steps to set the system to boot automatically.

Table 3–14 Setting the System to Boot Automatically

Step	Action	Result
1	Access the Setup menu.	The system displays the Setup menu.
2	Choose `Setup autoboot` and press Enter.	The system displays a prompt asking you whether you want the system to boot automatically.
3	If you do not want the system to boot automatically, choose No and press Enter.	The system displays the Setup menu.
4	If you want the system to boot automatically, choose Yes and press Enter.	The system prompts you to enter the countdown value. You specify the countdown value in seconds. The countdown value gives you that number of seconds to stop the system from booting.
5	Enter a value for the countdown and press Enter.	The system displays the Setup menu.
6	When you have finished making changes in the Setup menu, save your system setup. See the section entitled ‘‘Saving the System Setup or Quitting the Setup Menu ’’.	The system saves the changes.

Setup Autoboot Example

Example 3–11 shows an example of setting the system to autoboot and setting the countdown value to 10 seconds (default).

Example 3–11 Setting Autoboot

```
Should the system autoboot:
   Yes
   No

Enter Countdown value (in seconds): 10
```

Effects of This Procedure

Table 3–15 describes how this procedure affects other system settings and the actions that you must take as a result of using the procedure.

Table 3–15 Effects of Setting the System to Boot Automatically

Effect	Suggested Action
Sets the AUTOLOAD environment variable to YES or NO.	None.
Sets the COUNTDOWN environment variable to a value in seconds.	None.
Causes the system to begin a boot countdown sequence when you access the Boot menu.	Use the up arrow or down arrow keys to choose another menu item to cancel the boot countdown and remain in the ARC firmware.

Editing the System Environment Variables

Summary

This section describes how to edit or create user-defined Microsoft Windows NT system environment variables.

Important Information

Caution

Editing the values of the default ARC firmware environment variables could result in corrupted data or make the system inoperable.

Use the procedure described here only to edit or create your own environment variables. If you accidentally delete or modify one of the default ARC firmware environment variables, immediately reset the value back to the default as described in the section ‘‘Resetting the System to Factory Defaults ,’’ later in this chapter.

Editing Environment Variables

Table 3–16 lists the steps to edit the environment variables.

Table 3–16 Creating or Editing User-Defined Environment Variables

Step	Action	Result
1	Access the Setup menu.	The system displays the Setup menu.
2	Choose `Edit environment variables` and press Enter.	The system displays the environment variable edit screen.
3	Use the up arrow key or down arrow key to select the name of an existing environment variable or enter a new environment variable name.	Pressing the up arrow key or down arrow key displays the environment variable names in the list.
4	If necessary, edit the environment variable name. Press Enter when the environment name is correct.	The system displays the current value of the environment variable, if it already exists.
5	If you want to update the environment variable value, edit or enter the environment variable value and press Enter.	The system includes the modified or new environment variable at the bottom of the list.
6	If you want to delete the environment variable, leave the value field blank and press Enter.	The system deletes the variable from the list.
7	Press the escape (ESC) key to return to the Setup menu.	The system displays the Setup menu.
8	When you have finished making changes in the Setup menu, save your system setup. See the section entitled ‘‘Saving the System Setup or Quitting the Setup Menu ,’’ later in this chapter.	The system saves the changes.

Example

Example 3–12 illustrates the addition of a user-defined environment variable. C: is defined as the search path to be used by the ARC firmware to locate SCSI devices.

Example 3–12 Creating a User-Defined Environment Variable

```
Use Arrow keys to select a variable, ESC to exit:

   Name: C:
   Value: scsi()disk(0)rdisk()partition(1)

Environment variables:
    CONSOLEIN=multi()key()keyboard()console()
    CONSOLEOUT=eisa()video()monitor()console()
    FWSEARCHPATH=scsi()disk(0)rdisk()partition(1)
    TIMEZONE=PST8PDT
    A:=eisa()disk()fdisk()
    FLOPPY2=N
    FLOPPY=1
    AUTOLOAD=YES
    COUNTDOWN=10
    KEYBOARDTYPE=0
    VERSION=4.xx
    ENABLEPCIPARITYCHECKING=FALSE
```

Effects of This Procedure

This procedure creates an environment variable that you use in other parts of the ARC firmware.

Default Environment Variables

Summary

This section describes the default ARC firmware environment variables.

Important Information

Caution

Do not edit or delete the default firmware ARC environment variables. Editing the values of the default ARC firmware environment variables could result in corrupted data or make the system inoperable.

Default Environment Variables

Table 3–17 lists and explains the default ARC firmware environment variables.

Table 3–17 ARC Firmware Environment Variables

Variable	Description
CONSOLEIN	The console input device. The default value is multi( )key( )keyboard( )console( ).
CONSOLEOUT	The console output device. The default value is eisa( )video( )monitor( )console( ).
FWSEARCHPATH	The search path used by the ARC firmware and other programs to locate particular files. The default value is the same as the SYSTEMPARTITION environment variable value.
TIMEZONE	The time zone in which the system is located. This variable accepts ISO/IEC9945-1 (POSIX) standard values.
A:	The default floppy drive. The default value is eisa( )disk( )fdisk( ).

(continued on next page)

Table 3–17 (Cont.) ARC Firmware Environment Variables

Variable	Description
FLOPPY	The capacity of the default floppy drive, either 1 (1.2 MB), 2 (1.44 MB), or 3 (2.88 MB).
FLOPPY2	The capacity of an optional second floppy drive, either N (not installed), 1, 2, or 3.
AUTOLOAD	The default startup action, either Y (boot) or N or undefined (remain in ARC firmware).
COUNTDOWN	The default time limit in seconds before the system boots automatically when AUTOLOAD is set to yes. The default value is 10.
KEYBOARDTYPE	The keyboard language. The default is U.S. (English).
ENABLEPCIPARITY CHECKING	When set to the default value (FALSE), disables parity checking on the PCI bus in order to prevent machine check errors that can occur if the PCI device has not properly set the parity on the bus. **Note** Refer to the `set pci_parity` SRM environment variable for more information.

Other Environment Variables

The operating system or other programs, for example, the ECU, may create either temporary or permanent environment variables for their own use. Do not edit or delete these environment variables.

Resetting the System to Factory Defaults

Summary

This section describes how to reset the system to the factory default settings.

Resetting to Factory Defaults

Table 3–18 lists the steps to reset the system to the factory default settings.

Table 3–18 Resetting to Factory Defaults

Step	Action	Result
1	Access the Setup menu.	The system displays the Setup menu.
2	Choose `Reset system to factory defaults` and press Enter.	The system displays a selection menu asking you to confirm your choice.
3	Choose NO and press Enter if you do not want to continue.	The system displays the Setup menu.
4	Choose YES and press Enter if you want to continue.	The system sets the default environment variables, sets the default configuration, creates at least one boot selection, and sets the AUTOLOAD environment variable to YES. It then displays the Setup menu.
5	When you have finished making changes in the Setup menu, save your system setup. See the section entitled ‘‘Saving the System Setup or Quitting the Setup Menu ,’’ later in this chapter.	The system saves the changes.

Effects of This Procedure

Resetting the system to factory defaults clears the EISA configuration information and clears any user-defined environment variables. You must run the ECU on the system before you try to boot the operating system. See your system documentation for information on using the ECU.

Caution

Choosing this option deletes all boot selections and environment variables and replaces them with factory default values. Use this option with caution.

Getting Help

Summary

This section describes how to get help on the ARC firmware.

ARC Firmware Help

The ARC firmware does not provide help on all the items contained in its menus. However, it provides general help on using the Setup menu.

Displaying Setup Help

Table 3–19 lists the steps to display the Setup menu help screen.

Table 3–19 Displaying the Setup Menu Help Screen

Step	Action	Result
1	Enter the ARC firmware by shutting down the operating system following the procedure described in the operating system documentation. Turn the system off and then on.	The system displays the ARC firmware Boot menu.
2	Choose `Supplementary menu...` from the Boot menu.	The system displays the Supplementary menu.
3	Choose `Set up the system...` from the Supplementary menu.	The system displays the Setup menu.
4	Choose `Help` from the Setup menu and press Enter.	The system displays the Setup menu help screen.

Setup Menu Example

Example 3–13 shows the Setup menu.

Example 3–13 Selecting the Help Menu Item

```
ARC Multiboot Alpha Version n.nn           Thursday, 6-10-1995
9:49 :50 PM
Copyright (c) 1993--1995  Microsoft Corporation
Copyright (c) 1993--1995  Digital Equipment Corporation

Setup menu:

        .
        .
        .

     Reset system to factory defaults

     Help
     Switch to OpenVMS or OSF console
     Supplementary menu, and do not save changes...

Use the arrow keys to select, then press Enter.
```

Help Example

Example 3–14 shows the Setup menu help screen.

Example 3–14 Setup Menu Help Screen

```
Do the following steps, in this order, to set up the system:

1. Set system time.
2. Set default environment variables.
3. Set default configuration.
4. Create at least one boot selection.
5. Setup autoboot, if desired.
6. Run the EISA configuration utility.

-> A menu item with an arrow represents a section of the NVRAM with a
problem. Select these items (in top to bottom order) to repair
the NVRAM before attempting to boot or install Windows NT.

"Reset system to factory defaults" does steps 2 -- 5 for a typical
system.

Home, End, Delete, and Backspace will help you edit strings.
The ESCape key returns from a menu, and aborts a sequence.

The firmware automatically reboots if the configuration is changed.

 Press any key to continue...
```

Saving the System Setup or Quitting the Setup Menu

Summary

This section describes how to exit from the Setup menu with or without saving the changes you make to the system setup.

Saving the System Setup

Table 3–20 lists the steps to save the system setup after you make changes to it.

Table 3–20 Saving the System Setup

Step	Action	Result
1	Choose `Supplementary menu, and save changes...` and press Enter.	The system saves the changes that you have made in the NVRAM and then displays the Supplementary menu.

Quitting Without Saving

Table 3–21 lists the steps to quit from the Setup menu without saving the changes that you made to the system setup.

Table 3–21 Quitting the Setup Menu

Step	Action	Result
1	Choose `Supplementary menu, and do not save changes...` and press Enter.	The system discards the changes that you have made and then displays the Supplementary menu.

4

Managing ARC Boot Selections

Introduction

This chapter describes how to manage the boot selections that the Microsoft Windows NT operating system uses when booting.

Note

If you are running either the Digital UNIX or OpenVMS operating system, you do not need to read this chapter.

In This Chapter

This chapter contains the following sections:

- Accessing the Boot Selections Menu
- Information on Boot Selection Variables
- Adding a Boot Selection
- Checking the Boot Selections for Errors
- Changing a Boot Selection
- Displaying the System Boot Selections
- Changing the Default Boot Selection
- Deleting a Boot Selection

Accessing the Boot Selections Menu

Summary

This section describes how to access the Boot selections menu.

Accessing the Boot Selections Menu

Table 4–1 lists the steps to access the Boot selections menu.

Table 4–1 Accessing the Boot Selections Menu

Step	Action	Result
1	If necessary, access the Setup menu.	The system displays the Setup menu.
2	Choose the `Manage boot selections` menu item and press Enter.	The system displays the Boot selections menu.

Boot Selections Menu Example

Example 4–1 shows the Boot selections menu.

Example 4–1 Boot Selections Menu

```
Alpha Firmware Version n.nn
Copyright (c) 1993--1995 Microsoft Corporation
Copyright (c) 1993--1995  Digital Equipment Corporation

Boot selections menu:

    Add a boot selection
    Change a boot selection
    Verify boot selections
    Delete a boot selection
    Dump boot selections
    Rearrange boot selections
    Setup menu...

Use the arrow keys to select, then press Enter.
```

Information on Boot Selection Variables

Summary

This section describes the ARC boot selection variables that the system uses when booting.

Boot Selection Variables

Table 4–2 lists and explains the boot selection variables.

Table 4–2 ARC Boot Selection Variables

Variable	Description
LOADIDENTIFIER	The name of the boot selection, for example, Windows NT. You can specify any string value. The firmware does not check the value of this string.
SYSTEMPARTITION	The ARC-defined device name for the partition containing the operating system loader (OSLOADER.EXE for Windows NT).
OSLOADER	The full file name of the operating system loader, including partition device name, directories, and file name.
OSLOADPARTITION	The ARC-defined device name for the partition containing the operating system kernel.
OSLOADFILENAME	The directory name of the root directory for the operating system directory tree, \WINNT35.
OSLOADOPTIONS	A string that is passed to the operating system at load time. This string can be used to modify operating system behavior. The firmware does not check the value of this string.

Adding a Boot Selection

Summary This section describes how to add a boot selection.

Adding a Boot Selection Table 4–3 lists the steps that you must follow to add a boot selection.

Table 4–3 Adding a Boot Selection

Step	Action	Result
1	If necessary, access the Boot selections menu.	The system displays the Boot selections menu.
2	Choose `Add a boot selection` and press Enter.	The system displays a series of selection menus and prompts relating to the Windows NT operating system.
3	If you have not changed the location of the operating system files, you can use the default answers and choices provided by the procedure.	The system includes the boot selection in the list of boot selections and then displays the Boot selections menu.
4	If you have changed the location of the operating system files, you must edit the default answers or choose selections other than those provided by the procedure. See your operating system documentation for more information.	The system includes the boot selection in the list of boot selections and then displays the Boot selections menu.
5	If you have finished modifying the boot selections, choose `Setup menu...` and press Enter.	The system displays the Setup menu.

Example

Example 4–2 shows an example of adding a boot selection.

Example 4–2 Adding a Boot Selection

```
 Select a system partition for this boot selection:
    Scsi Hard Disk 0 Partition 1
    New system partition

 Enter location of system partition for this boot selection:
      Select media:
    Scsi Hard Disk
    Floppy Disk
    CD-ROM

      Enter SCSI ID: 4

 Enter the osloader directory and
name: \os\nt\osloader.exe

 Is the operating system in the same partition as the osloader:
    Yes
    No

 Enter location of os partition:
      Select media:
    Scsi Hard Disk
    Floppy Disk
    CD-ROM

      Enter SCSI ID: 0
      Enter partition : 1

 Enter the operating system root directory: \winnt

 Enter a name for this boot selection: Windows NT
Do you want to initialize the debugger at boot time:
    Yes
    No
```

Checking the Boot Selections for Errors

Summary

This section describes how to check the system boot selections for errors.

Errors in Boot Selections

There are two main causes of errors in boot selections:

- **The boot selection is defined incorrectly.**

  This causes the system to search in the wrong locations for the operating system files.

- **The operating system is installed incorrectly.**

  This causes the system to fail when it tries to locate the operating system files.

You must consider both possibilities when trying to correct any errors detected in the boot selection.

Checking the Boot Selections

Table 4–4 lists the steps that you must follow to check the boot selections for errors.

Table 4–4 Checking the Boot Selections for Errors

Step	Action	Result
1	If necessary, access the Boot selections menu.	The system displays the Boot selections menu.
2	Choose `Check boot selections` and press Enter.	The system checks the boot selections for errors. If it does not detect any errors, it remains in the Boot selections menu. If it does detect errors, it displays the name of the boot selection in which it detected the error, a selection menu, and a message indicating the cause of the error.

(continued on next page)

Table 4–4 (Cont.) Checking the Boot Selections for Errors

Step	Action	Result
3	If you want to change the boot selection variables causing the error, choose the `Change this boot selection` menu item and press Enter. Follow the steps listed in Table 4–5 starting at step 4 for information on changing the variable values.	The system displays the Boot selection edit display, listing the boot selection variables and their values.
4	If you want to delete the boot selection, choose `Delete this boot selection` and press Enter.	The system deletes the boot selection. If the system detects more boot selections with errors, it repeats the display for that boot selection. Otherwise, it displays the Boot selections menu.
5	If you want to ignore errors in the boot selection, choose `Ignore problems with this boot selection` and press Enter.	If the system detects more boot selections with errors, it repeats the display for that boot selection. Otherwise, it displays the Boot selections menu. When you return to the Setup menu, the `Manage boot selections` menu item error indicator (->) remains.
6	If you have finished modifying the boot selections, choose `Setup menu...` and press Enter.	The system displays the Setup menu.

Example

Example 4–3 shows an example of the display that results when the system detects an error in a boot selection.

Example 4–3 Checking the Boot Selections for Errors

```
Problems were found with Windows NT. Choose an action:

   Ignore problems with this boot selection
   Delete this boot selection
   Change this boot selection

OSLOADPARTITION cannot be found, value is:
    scsi()disk(0)rdisk()partition(2)
```

Changing a Boot Selection

Summary

This section describes how to change a boot selection by modifying the values of the boot selection variables.

Changing a Boot Selection

Table 4–5 lists the steps to change the value of a boot selection variable.

Table 4–5 Changing a Boot Selection

Step	Action	Result
1	If necessary, access the Boot selections menu.	The system displays the Boot selections menu.
2	Choose `Change a boot selection` and press Enter.	The system displays a selection menu listing the names of the system boot selections.
3	Choose the name of the boot selection that you want to change and press Enter.	The system displays the Boot selection edit display, listing the boot selection variables and their values.
4	Press the up arrow key or down arrow key to select the name of a variable.	Pressing the up arrow key or down arrow key displays the boot selection variable names in the list.
5	Press Enter when the system displays the name of the variable whose value you want to change.	The system displays the current value of the variable.
6	To update the boot selection variable value, edit the existing value and press Enter.	The system includes the modified or new boot selection variable in the boot selection display.
7	Press the escape (ESC) key to return to the Boot selections menu.	The system displays the Boot selections menu.
8	If you have finished modifying the boot selections, choose `Setup menu...` and press Enter.	The system displays the Setup menu.

Example

Example 4–4 shows an example of the display used for changing a boot selection.

Example 4–4 Changing a Boot Selection

```
 Use Arrow keys to select a variable, ESC to exit:

    Name:  OSLOADFILENAME
    Value:  \winnt

Environment variables for boot selection 1:
     LOADIDENTIFIER=Windows NT
     OSLOADPARTITION=scsi()disk(0)rdisk()partition(2)
     OSLOADER=scsi()disk(0)rdisk()partition(1)\osloader.exe
     SYSTEMPARTITION=scsi()disk(0)rdisk()partition(1)
     OSLOADFILENAME=\winnt
     OSLOADOPTIONS=nodebug
```

Displaying the System Boot Selections

Summary

This section describes how to display all the system boot selections.

Displaying the System Boot Selections

Table 4–6 lists the steps to display all the system boot selections.

Table 4–6 Displaying the System Boot Selections

Step	Action	Result
1	If necessary, access the Boot selections menu.	The system displays the Boot selections menu.
2	Choose `Dump boot selections` and press Enter.	The system displays the system boot selections. The boot selection variable values for each boot selection are separated by semicolons (;). When possible, the values share the same line to save space on the screen.
3	When you have finished examining the boot selections, press Enter or any other key.	The system displays the Boot selections menu.

Example

Example 4–5 shows an example of a boot selections display.

Example 4–5 System Boot Selections Display

```
LOADIDENTIFIER=Windows NT 2;Windows NT ;Windows NT 3
SYSTEMPARTITION=scsi()disk(0)rdisk()partition(1);
                scsi()disk(1)rdisk()partition(1);
                scsi()disk(0)rdisk()partition(1)
OSLOADER=scsi()disk(0)rdisk()partition(1)\os\nt\osloader.exe;
         scsi()disk(0)rdisk()partition(1)\os\nt\osloader.exe;
         scsi()disk(0)rdisk()partition(1)\osloader.exe
OSLOADPARTITION=scsi()disk(0)rdisk()partition(1);
                scsi()disk(0)rdisk()partition(1);
                scsi()disk(0)rdisk()partition(2)
OSLOADFILENAME=\winnt;\winnt;\winnt
OSLOADOPTIONS=nodebug;nodebug;nodebug

 Press any key to continue...
```

Changing the Default Boot Selection

Summary

This section describes how to change the default boot selection.

Default Boot Selection Definition

The default boot selection appears at the top of the system boot selections list. It is the boot selection that the system attempts to boot automatically if the AUTOLOAD environment variable is set to YES.

Changing the Default Boot Selection

Table 4–7 lists the steps to change the default boot selection.

Table 4–7 Changing the Default Boot Selection

Step	Action	Result
1	If necessary, access the Boot selections menu.	The system displays the Boot selections menu.
2	Choose `Rearrange boot selections` and press Enter.	The system displays a selection menu listing the names of the system boot selections.
3	If you do not want to change the default boot selection, press the escape (ESC) key.	The system displays the Boot selections menu without changing the default boot selection.
4	If you want to change the default boot selection, choose the name of the boot selection that you want to become the default boot selection and press Enter.	The system places that boot selection at the top of the list, and it becomes the default boot selection.
5	Press the escape (ESC) key when you have finished.	The system displays the Boot selections menu.
6	Choose `Setup menu...` and press Enter.	The system displays the Setup menu.

Example

Example 4–6 shows an example of changing the default boot selection. When you press Enter on the highlighted selection (Windows NT 3), it moves to the top and becomes the default.

Example 4–6 Changing the Default Boot Selection

```
Pick selection to move to the top, ESC to exit:
   Windows NT
   Windows NT 2
   Windows NT 3
```

Deleting a Boot Selection

Summary This section describes how to delete a boot selection.

Deleting a Boot Selection Table 4–8 lists the steps to delete a boot selection.

Table 4–8 Deleting a Boot Selection

Step	Action	Result
1	If necessary, access the Boot selections menu.	The system displays the Boot selections menu.
2	Choose `Delete a boot selection` and press Enter.	The system displays a selection menu listing the names of the system boot selections.
3	If you do not want to delete a boot selection, press the escape (ESC) key.	The system then displays the Boot selections menu without deleting a boot selection.
4	If you want to delete a boot selection, choose the name of that boot selection and press Enter.	The system deletes the boot selection and displays the Boot selections menu.
5	If you have finished modifying the boot selections, choose `Setup menu...` and press Enter.	The system displays the Setup menu.

Example Example 4–7 shows an example of deleting a boot selection.

Example 4–7 Deleting a Boot Selection

```
Selection to delete:
   Windows NT
   Windows NT 2
   Windows NT 3
```

Index

A

A: environment variable (ARC), 3–38
Alternate operating system, booting (ARC), 3–14
arc command (SRM), 1–14
ARC firmware
  accessing the Boot menu, 3–10
  adding a boot selection, 4–4
  boot device names, listing, 3–21
  Boot menu functions, 3–5
  Boot selections menu, 4–2
  booting default boot selection, 3–12
  changing boot selections, 4–8
  changing the default boot selection, 4–12
  configuring floppy drives, 3–30
  default environment variables, 3–35, 3–38
  deleting a boot selection, 4–14
  device names, 3–16
  discarding Setup menu changes, 3–45
  displaying boot selections, 4–10
  editing environment variables, 3–35
  hardware configuration display, 3–22
  help, 3–43
  interface menus, 3–1
  introduction, 3–5
  invoking from SRM firmware, 1–14
  keyboard and keys, 3–6
  monitor program, 3–5
  resetting to defaults, 3–40
  running a program from, 3–16
  saving Setup menu changes, 3–45
  setting default system partition, 3–27
ARC firmware (cont'd)
  setting time and date, 3–25
  Setup menu, 3–19, 3–43
  Setup menu functions, 3–6
  Supplementary menu, 3–17
  Supplementary menu functions, 3–5
  updating, 3–18
  using, 3–1
ARC interface
  help menu, 3–42
  key functions, 3–6
Autoboot
  Digital UNIX, 1–33
  OpenVMS, 1–33
  Windows NT (ARC), 3–33
AUTOLOAD environment variable (ARC), 3–28, 3–33, 3–39
auto_action environment variable (SRM), 1–31

B

Baud rate, changing, 1–40
boot command (SRM), 1–15
boot command, in secure mode, 2–7
Boot device, default
  setting for Digital UNIX, 1–34
  setting for OpenVMS, 1–34
  setting for Windows NT, 4–12
Boot devices
  ARC firmware names listing, 3–21
  in hardware configuration display, 3–22
Boot examples (SRM), 1–16

Boot flags (SRM), 1–16
Boot menu (ARC)
accessing, 3–10
functions, 3–5
Boot procedure
ARC boot selections, 4–1
booting alternate operating system (ARC), 3–14
booting default boot selection (ARC), 3–12
booting Digital UNIX automatically, 1–33
booting OpenVMS automatically, 1–33
booting Windows NT automatically, 3–33
Boot selection variables (ARC)
displaying, 4–10
editing, 4–8
LOADIDENTIFIER, 4–3
OSLOADER, 4–3
OSLOADFILENAME, 4–3
OSLOADOPTIONS, 4–3
OSLOADPARTITION, 4–3
SYSTEMPARTITION, 4–3
Boot selections (ARC)
adding, 4–4
alternate, booting, 3–14
changing, 4–8
checking for errors, 4–6
default, booting, 3–12
default, changing, 4–12
deleting, 4–14
displaying, 4–10
managing, 4–1
Boot selections menu (ARC)
accessing, 4–2
bootdef_dev environment variable (SRM), 1–31
Bootstrap
command (SRM), 1–15
examples (SRM), 1–16
flags (SRM), 1–16
boot_osflags environment variable (SRM), 1–31
Bus probe algorithm, setting, 1–39
Buses, displaying (SRM), 1–75
bus_probe_algorithm environment variable (SRM), 1–31

C

cdp command (SRM), 1–17
Changes
discarding in ARC firmware, 3–45
saving in ARC firmware, 3–45
Characters, control, supported in console mode, 1–8
Characters, keyboard, supported in console mode, 1–8
clear password, security command, 2–6
com1_baud environment variable (SRM), 1–31
Compact disc drive
running Windows NT programs from, 3–16
Configuration capacities
floppy drive, 3–31
Console action
setting at power-up, 1–33
Console commands
arc, 1–14
boot, 1–15
cdp, 1–17
continue, 1–20
ecu, 1–21
edit, 1–13, 1–22
help, 1–24
init, 1–25
language, 1–52
man, 1–24
more, 1–28
set, 1–29
set auto_action, 1–33
set bootdef_dev, 1–34
set boot_osflags, 1–36
set bus_probe_algorithm, 1–39
set com1_baud, 1–40
set console, 1–41
set controlp, 1–42
set cpu_enabled, 1–43

Console commands (cont'd)
set ew*0_mode, 1–45
set ew*0_protocols, 1–47
set host, 1–49
set kbd_hardware_type, 1–51
set ocp_text, 1–54
set os_type, 1–55, 3–10
set pci_parity, 1–57
set pci_read_mult, 1–58
set pk*0_fast, 1–59
set pk*0_host_id, 1–61
set pk*0_soft_term, 1–62
set tga_sync_green, 1–64
set tt_allow_login, 1–65
show, 1–66
show config, 1–68
show device, 1–75
show fru, 1–81
show memory, 1–85
show pal, 1–86
show version, 1–87
test, 1–88
Console commands (SRM)
displaying information about, 1–24
editing, 1–9
extending to a second line, 1–9
format, 1–12
levels, 1–10
list of, 1–10
online help, 1–12
recalling, 1–9
Console editor (SRM), 1–13
console environment variable (SRM), 1–31
Console firmware
ARC, updating, 3–18
Digital UNIX/osf_1, 1–3
OpenVMS, 1–3
operating system support, 1–2
overview, 1–2
services, 1–2
setting default, 1–55
updating, x
Windows NT, 1–2
Console interfaces
switching between, 1–7, 3–4
Console mode
keyboard characters, 1–8
supported control characters on VT-series terminals, 1–8
Console modes
secure mode, 2–2
user mode, 2–2
CONSOLEIN environment variable (ARC), 3–38
CONSOLEOUT environment variable (ARC), 3–38
continue command (SRM), 1–20
Control characters
use of with VT-series terminal, 1–8
Controllers, displaying (SRM), 1–75
controlp environment variable (SRM), 1–42
COUNTDOWN environment variable (ARC), 3–28, 3–33, 3–39
cpu_enabled environment variable (SRM), 1–31
Ctrl/x command, secure mode, 2–8

D

Date
setting for ARC firmware, 3–25
Default ARC boot
selection, booting, 3–12
selection, changing, 4–12
Default boot device
setting (SRM), 1–34
Default boot flags
setting (SRM), 1–36
Default system partition
setting from ARC interface, 3–27
Defaults
resetting ARC firmware to, 3–40
Device names
ARC firmware, 3–16
Device naming convention, 1–75
Devices
ARC firmware boot names listing, 3–21
displaying (SRM), 1–75

Devices (cont'd)
in hardware configuration display, 3–22
Disabling CPUs, 1–43
DUP server, connecting console program to, 1–49

E

ECU
invoking, 1–21
ecu command (SRM), 1–21
edit command (SRM), 1–22
edit command, (SRM), 1–13
Editing RAM files or nvram file, 1–22
EISA Configuration Utility (ECU), 1–5, 3–1
EISA devices, displaying, 3–21
ENABLEPCIPARITYCHECKING environment variable (ARC), 3–39
Enabling CPUs, 1–43
Environment variables
default ARC, 3–38
Environment variables (ARC)
See also Boot selection variables
A:, 3–38
AUTOLOAD, 3–28, 3–33, 3–39
CONSOLEIN, 3–38
CONSOLEOUT, 3–38
COUNTDOWN, 3–28, 3–33, 3–39
creating, 3–35
editing for Windows NT, 3–35
ENABLEPCIPARITYCHECKING, 3–39
FLOPPY, 3–30, 3–38
FLOPPY2, 3–30, 3–39
FWSEARCHPATH, 3–28, 3–38
KEYBOARDTYPE, 3–39
other, 3–39
resetting default variables, 3–35
SYSTEMPARTITION, 3–29
TIMEZONE, 3–38
Environment variables (SRM)
list of, 1–31
preset values, 1–31
setting or modifying, 1–29
value, displaying, 1–66
Errors
checking the ARC boot selections, 4–6
commands to report, 1–81
Ethernet option, configuring (SRM), 1–45
ew*0_protocols environment variable (SRM), 1–31
Exiting ARC Setup menu, 3–45

F

Factory defaults
resetting for Windows NT, 3–40
FAT file format, 3–16
File formats
FAT, 3–16
for running programs, 3–16
NTFS, 3–16
Firmware version
displaying information for (SRM), 1–87
Floppy disk drive
capacities, 3–31
configuring for Windows NT, 3–30
FLOPPY environment variable (ARC), 3–30, 3–38
running Windows NT programs from, 3–16
FLOPPY2 environment variable (ARC), 3–30, 3–39
Formats, hardware configuration display, 3–21
FRUs
commands to report errors, 1–81
FWSEARCHPATH environment variable (ARC), 3–28, 3–38

G

Graphics accelerator, ZLXp-E, 1–64
Graphics monitor, synchronizing, 1–64

H

Hardware configuration display, ARC, 3–22
Hardware configuration, displaying, 3–21
Hardware devices
  See Devices
Help
  accessing in console mode (SRM), 1–12
  on ARC firmware, 3–43
help command (SRM), 1–24
Help menu item
  ARC firmware, 3–43
Help screen
  on ARC firmware Setup menu, 3–43

I

init command (SRM), 1–25
Initialization, system, 1–25

K

kbd_hardware_type environment variable (SRM), 1–31
Keyboard
  ARC firmware keys, 3–6
Keyboard characters, supported in console mode, 1–8
KEYBOARDTYPE environment variable (ARC), 3–39

L

language environment variable (SRM), 1–31
Language, setting default, 1–52
LOADIDENTIFIER boot selection variable (ARC), 4–3
Login, on alternative ports, 1–65
login, security command, 2–4

M

man command (SRM), 1–24
Memory module
  displaying information for (SRM), 1–85
Monitor
  ARC keyboard keys, 3–6
more command (SRM), 1–28

N

Network protocols, enabling (SRM), 1–47
NTFS file format, 3–16
nvram power-up script, 1–13

O

ocp_text environment variable (SRM), 1–31
Online help
  in console mode (SRM), 1–12
Operating system
  setting default, 1–55
OSLOADER boot selection variable (ARC), 4–3
OSLOADFILENAME boot selection variable (ARC), 4–3
OSLOADOPTIONS boot selection variable (ARC), 4–3
OSLOADPARTITION boot selection variable (ARC), 4–3
os_type environment variable (SRM), 1–31

P

PALcode version
  displaying information for (SRM), 1–86
Parity checking, disabling, 1–57
Partitions, system
  setting default from ARC interface, 3–27
pci_parity environment variable (SRM), 1–31
pci_read_mult environment variable (SRM), 1–31

pk*0_soft_term environment variable (SRM), 1–31
Power-up output, displaying (SRM), 1–41
Power-up script, 1–13
Power-up/diagnostic display
  setting message on (SRM), 1–54
Program, resuming, 1–20
Program, Windows NT
  running, 3–16
  valid ARC firmware programs, 3–16

Q

QLogic ISP1020 controller, enabling or disabling (SRM), 1–62
Quitting ARC Setup menu, 3–45

R

RAID Standalone Configuration Utility, 1–5, 3–1
Reset
  Windows NT factory defaults, 3–40
Root number, 1–37

S

SCSI controller bus node ID
  setting default (SRM), 1–61
SCSI controller, QLogic ISP1020, enabling or disabling (SRM), 1–62
SCSI devices
  in hardware configuration display, 3–22
SCSI devices, displaying, 3–21
SCSI, fast
  configuring (SRM), 1–59
Secure mode, 2–2
Secure mode functions, 2–7
Security commands
  clear password, 2–6
  login, 2–4
  set password, 2–3
  set secure, 2–4
set auto_action command (SRM), 1–33
set bootdef_dev command (SRM), 1–34
set boot_osflags command (SRM), 1–36
  boot flags argument (Digital UNIX), 1–36
  boot flags argument (OpenVMS), 1–36
  root number argument (Digital UNIX), 1–36
  root number argument (OpenVMS), 1–36
set bus_probe_algorithm command (SRM), 1–39
set com1_baud command (SRM), 1–40
set com2_baud command (SRM), 1–40
set command (SRM), 1–29
set console command (SRM), 1–41
set controlp command (SRM), 1–42
set cpu_enabled command (SRM), 1–43
set ew*0_mode command (SRM), 1–45
set ew*0_protocols command (SRM), 1–47
set host command (SRM), 1–49
set kbd_hardware_type command (SRM), 1–51
set language command (SRM), 1–52
  supported languages, 1–53
set ocp_text command (SRM), 1–54
set os_type command (SRM), 1–55
set password, security command, 2–3
set pci_parity command (SRM), 1–57
set pci_read_mult command (SRM), 1–58
set pk*0_fast command (SRM), 1–59
set pk*0_host_id command (SRM), 1–61
set pk*0_soft_term command (SRM), 1–62
set secure, security command, 2–4
set tga_sync_green command (SRM), 1–64
set tt_allow_login command (SRM), 1–65
Setup menu, ARC, 3–43
  accessing, 3–19
  discarding changes, 3–45
  example, 3–20
  exiting or quitting, 3–45
  functions, 3–6
  saving changes, 3–45
show command (SRM), 1–66
show config command (SRM), 1–68

show device command (SRM), 1–75
show fru command (SRM), 1–81
show memory command (SRM), 1–85
show pal command (SRM), 1–86
show version command (SRM), 1–87
SRM console commands
See Console commands (SRM)
SRM interface
switching to, 1–7, 3–4
switching to ARC from, 1–7, 3–4
Standalone Configuration Utility, 1–5, 3–1
start and continue commands, secure mode, 2–7
Supplementary menu, ARC
accessing, 3–17
example, 3–17
functions, 3–5
System
resetting to ARC firmware defaults, 3–40
System date and time
setting from ARC interface, 3–25
System devices
ARC firmware boot device names, 3–21
System initialization, 1–25
System partition
defined, 3–27
setting from ARC interface, 3–27
System setup, Windows NT
quitting without saving, 3–45
saving, 3–45
System, testing, 1–88
SYSTEMPARTITION boot selection variable (ARC), 4–3
SYSTEMPARTITION environment variable, 3–29

T

Terminal
ARC firmware keys, 3–6
test command (SRM), 1–88
Testing system, 1–88
tga_sync_green environment variable (SRM), 1–31
Time
setting for ARC firmware, 3–25
TIMEZONE environment variable (ARC), 3–38
tt_allow_login environment variable (SRM), 1–31

U

User mode, 2–2

W

Windows NT
booting alternate operating system, 3–14
booting automatically, 3–33