StorageWorks™

RAID Array 310
Configuration and Maintenance Guide

User's Guide

Order Number: EK–SMCS2–UG. B01

Digital Equipment Corporation
Maynard, Massachusetts

Second Edition March 1997

Printed in U. S. A

Contents

3 Configuring Your Subsystem (cont.)

4 Maintaining the RAID Array 310 Subsystem

4 Maintaining the RAID Array 310 Subsystem (cont.)

5 Monitoring Subsystem Performance

5 Monitoring Subsystem Performance (cont.)

6 Maintaining Devices in RAID Array 310 Subsystems

7 Troubleshooting

A RAID Array 310 CLI Commands

A RAID Array 310 CLI Commands (cont.)

Tables

Tables (cont.)

Figures

Revision Record

This Revision Record provides a concise publication history of this manual. It lists the manual revision levels, release dates, and reasons for the revisions. It also describes how the changes to affected pages are marked in the manual.

The following revision history lists all revisions of this publication and their effective dates. The publication part number is included in the Revision Level column, with the last entry denoting the latest revision. This publication supports the StorageWorks RAID Array 310 Subsystem.

Revision Level	Date	Summary of Changes
EK–SMCS2–UG. A01	March 1996	Initial Release
EK–SMCS2–UG. B01	March 1997	Upgraded to HSOF Version 3.1

About This Guide

This section identifies the audience of this guide and describes the contents (chapter by chapter) and structure. In addition, this section includes a list of associated documents and the conventions used in this guide.

This guide provides the following:

- How to configuring the RAID Array 310 using the onboard controller firmware.
- How to maintain the RAID Array 310 using the onboard controller firmware

Audience

This guide is intended for administrators and system integrators of StorageWorks RAID Array 310 Subsystems. Configuring the StorageWorks RAID Array 310 Subsystem requires a general understanding of RAID concepts and storage concepts.

NOTE

The onboard controller firmware makes reference to failover, dual-redundant controllers and other language associated with the use of two controllers. However, the RAID Array 310 Subsystem supports only single controller use.

Document Structure

This guide contains the following chapters:

Chapter 1: Introduction

This chapter provides a general description of what's involved with configuring and maintaining the RAID Array 310 Subsystem through the onboard controller firmware. This chapter also provides a configuration roadmap and identifies where you can locate much of the information throughout this document.

Chapter 2: Using the RAID Array 310 Command Line Interface (CLI)

This chapter describes the components of the RAID Array 310 onboard controller firmware Command Line Interface (CLI) and how to access it.

Chapter 3: Configuring the Subsystem

This chapter describes the steps required to configure a RAID Array 310 Subsystem for the first time. In addition, the chapter describes the type of storagesets you can create how to setup for automatically replenishing the supply of spare devices in the subsystem and explains concepts such as transportability and chunksize.

Chapter 4: Maintaining the RAID Array 310 Subsystem

This chapter describes how to maintain the RAID Array 310 Subsystem by viewing subsystem configuration and states, recovering after a device failure, maintaining the spareset, changing storageset parameter settings, removing member devices from storagesets, upgrading the controller firmware, and setting feature licenses.

Chapter 5: Monitoring Subsystem Performance

This chapter describes how to use the following local program utilities to monitor subsystem performance: Fault Management Utility (FMU), VTDPY Utility, and DILX Utility.

Chapter 6: Maintaining Devices in RAID Array 310 Subsystems

This chapter describes how to maintain devices in RAID Array 310 Subsystems by matching devices displayed on the display with their counterpart in the subsystem cabinet, formatting a SCSI device and updating a device's firmware.

Chapter 7: Troubleshooting

This chapter describes what to do to troubleshoot the RAID Array 310 Subsystem, how to isolate faults directly and through CLI messages and controller error codes, how to avoid unwanted unwritten cached data, and how to use CLI commands with write back cache.

Appendix A: The RAID Array 310 CLI Commands

Appendix A describes each of the CLI commands available for the RAID Array 310 Subsystem including syntax, parameters and examples of how to use them.

Associated Documents

In addition to this guide, the following documentation is useful to the reader:

Table 1 Associated Documents

Document Title	Order Number
RAID Array 310 Deskside Subsystem User's Guide	EK-SMCPL-UG
Getting Started - RAID Array 310 for Solaris	EK-SMRA8-UG
Getting Started - RAID Array 310 for Digital UNIX	EK-SMRA9-UG

Conventions

This guide uses the following documentation conventions:

Table 2 Style Conventions

Style	Meaning
`boldface monospace type`	To be input by the user
`plain monospace type`	Screen text
italic type	For emphasis, manual titles, utilities, menus, screens, and filenames

In addition, although the onboard controller firmware makes reference to failover, dual-redundant controllers and other language associated with the use of two controllers, the RAID Array 310 Subsystem supports single controller use only.

1

Introduction

This chapter describes the procedures required to maintain your subsystem online through the firmware on the controller. In addition, this chapter describes the areas of the firmware with which you interact, the Command Line Interface (CLI) and related utilities.

The RAID Array 310 subsystem provides a RAID storage solution with up to seven storage devices.

You set various aspects of the operation of the subsystem and get information about the status of the subsystem through firmware onboard the controller. Operations you can perform and information you can get with the onboard firmware include:

- Setting data storage configuration
- Changing data storage configuration
- Setting general controller operating parameters
- Downloading firmware to SCSI devices
- Troubleshooting the Subsystem
- Subsystem Status

The specific areas of the onboard controller firmware with which you work include the Command Line Interface (CLI) facility and local utility and diagnostic programs. Chapter 2 describes the CLI facility and local utility and diagnostic programs. Appendix A details the use of each of the CLI commands.

1.1 Configuring the RAID Array 310 Subsystem

Configure the RAID Array 310 Subsystem with any of the data storage methods described in Table 1–1.

Table 1–1 Configuration Options

Storage Method	Type of storageset	Number of Devices Required for this Storage Method
Individual Devices	Disk drive	1 - 7
RAID 0	Stripeset	2 – 7 devices
RAID 1	Mirrorset	2 – 3 devices per mirror set, up to 3 mirror sets per subsystem
RAID 0 + 1	Striped mirrorsets	2 – 3 mirror sets
RAID 3/5 A redundant-stripeset combining the optimized data transfers of RAID 3 with the striping of parity of RAID 5.	RAIDset	3 - 7 devices

Follow the procedures described in Chapter 3 to configure the RAID Array 310 Subsystem. Table 1–2 displays the major steps involved in configuring.

Table 1–2 Configuration Steps

Steps	In this Guide
Set the controller ID as a target on the SCSI bus.	See Chapter 3, Section 3.1
Tell the controller what devices are available to it.	See Chapter 3, Section 3.2
Configure the subsystem by selecting the storagesets you want to create, then creating them with the devices installed in your subsystem cabinet.	See Chapter 3, Sections 3.3–3.6
Save copies of the configuration on devices or storagesets.	See Chapter 3, Section 3.6.7.3
Tell the host what device resources are available on the controller by naming each device or storageset as a unit.	See Chapter 3, Section 3.7

Use the following guidelines to configure the RAID Array 310 subsystem to achieve optimal performance:

- Configure your RAID Array 310 to contain multiple devices or storagesets rather than one.
- Make use of the controller's two ports (channels). When you add devices to the controller, add some of the devices and storagesets to each of the controller ports. Distributing your devices and storagesets over both controller ports allows parallel activities to occur through the controller.
- Distribute your device or storageset units across the four possible target SCSI IDs for the controller. Specify a unique target SCSI ID for the first four device or storageset units that you configure. Any additional device or storageset units will need to share one of the target SCSI IDs; however, the load will already be fairly balanced.
- Avoid configuring multiple mirrorsets with the first member on the same port. For example:

  ```
  add mirrorset m1 disk100 disk200
  add mirrorset m1 disk210 disk110
  ```
- For write performance dependent applications, turn **on** the writeback cache. (The writeback cache is turned off by default.)
- Set the size of the maximum_cache_transfer of each unit to the largest size allowable, 1024, so that all transfers get cached. Otherwise, transfers that exceed the maximum_cache_transfer size would not get cached.

1.2 Changing the RAID Array 310 Subsystem Configuration

You can change the RAID Array 310 subsystem configuration, if your data storage needs change, by implementing a new data storage method in place of an existing one. You can also change the RAID Array 310 subsystem configuration by adjusting characteristics of the operation of an existing data storage method.

Implementing a new storage method generally involves backing up the data on your subsystem, undoing an existing storage method, then implementing a new one. Chapter 3 of this guide describes how to create and delete storage configurations.

Chapter 4 describes which characteristics of each of the storage methods you can change and how to go about making those changes.

1.3 Maintaining the Subsystem

Maintaining your subsystem through the onboard controller firmware involves monitoring the status of the subsystem and troubleshooting performance degradation and errors through:

- Viewing the Subsystem status
- Viewing error logs
- Analyzing subsystem performance
- Creating a snapshot of host data
- Adding devices to the spareset
- Deleting devices from the failedset

Maintaining your subsystem may also involve upgrading the controller firmware.

Chapter 4 describes the various methods available for monitoring and analyzing the RAID Array 310 subsystem performance. Chapter 4 also describes the methods of upgrading controller firmware.

Chapter 7 describes problems that may occur during operation of the RAID Array 310 Subsystem and how to correct them.

1.4 Maintaining Devices in the RAID Array 310 Subsystem

Maintaining devices in the RAID Array 310 subsystem through the onboard controller firmware involves the following:

- Locating devices displayed on the terminal or host system connected to the RAID Array 310 controller in the subsystem cabinet
- Formatting SCSI Devices
- Updating a SCSI Device's firmware

Chapter 6 describes the procedures for maintaining devices in the RAID Array 310 subsystem.

2

Using the RAID Array 310 Command Line Interface

This chapter describes the Command Line Interface (CLI) of the RAID Array 310 onboard controller firmware, its command sets, the local utility programs available on the controller firmware, and the basics on how to use the CLI.

2.1 Using the Command Line Interface (CLI)

The Command Line Interface (CLI) facility provides a series of commands that allow you to interact with the controller's firmware to change and adjust characteristics of the subsystem configuration and performance. The CLI also allows you to startup local programs – using the CLI RUN command – that reside in the controller firmware. You access the CLI by connecting a maintenance terminal or host virtual terminal interface to the controller.

2.1.1 The CLI Commands

The commands available through the controller firmware CLI allow you to control various aspects of the controller and subsystem operation. The CLI commands control either the controller, devices, storagesets or logical units. Table 2–1 provides a general description of each of the command sets.

Table 2–1 The CLI Command Sets

Command Set	Functions
Controller Commands	• Set and show basic controller parameters • Set and show controller ID (SCSI target ID). • Set the resident terminal characteristics. • Restart the controller. • Run resident diagnostics and utilities
Device Commands	• Specify and show the location of *physical* SCSI–2 devices attached to the controller. Each device's SCSI port-target-LUN (PTL) designation specifies its locations • Set and show the basic controller parameters.
Storageset Commands (Storagesets bind together devices and allow them to be manipulated as single units. The types of storagesets include: stripesets, raidsets, mirrorsets, and the spareset, and failedset.)	• Add, modify, rename, and show storagesets (stripesets, mirrorsets, and RAIDsets). • These commands also apply (to some extent) to the spareset and failedset. • Display the basic controller parameters.
Logical Unit Commands	• Add, modify, and show logical units built from devices and storagesets. • Add, modify, rename, and show storagesets (stripesets, mirrorsets, and RAIDsets). • Provides access to switches to control and optimize access to logical units of certain device types.

A description of each CLI command including, syntax and example of how to use it, appears in Appendix A.

2.1.2 Local Utility Programs

The firmware in the RAID Array 310 controller contains a number of local utility programs to aid you in monitoring the subsystem, changing the configuration and optimizing the performance of your subsystem. You access these programs using the CLI RUN command.

CAUTION

If you enter a DIRECTORY command at the CLI prompt, a utility entitled "Crash" is listed. The Crash utility is used for firmware debugging, and is not a user utility. The Crash utility causes a controller restart, and may result in the loss or corruption of data. Do not use the Crash utility.

The controller firmware contains the local utility programs described in Table 2–2.

Table 2–2 The Firmware Local Utility Programs

Local Utility Program	Description
CFMENU	Use the configuration menu (CFMENU) to quickly configure storage devices attached to the controller and to modify your subsystem configuration with a user-friendly menu interface.
CONFIG	Use the configuration (CONFIG) utility to allow the utility to automatically add the devices that exist in the storage subsystem for you by checking the controller’s device ports for unconfigured devices and adding them to the system.
CHVSN	Use the change volume serial number (CHVSN) utility to view and change storage device volume serial numbers.
CLCP	Use the code load/code patch (CLCP) utility to update the firmware contained in the PCMCIA card by downloading it via the maintenance port or host interface. CLCP also lets you install program patches in the firmware.
CLONE	Use the CLONE utility to partially automate the process of creating a snapshot of host unit data in the form of a mirrorset.
FLS	Use the Firmware Licensing System (FLS) utility to control such value-added, licensed, controller features as RAID and write-back caching.
FMU	Use the Fault Management Utility (FMU) to customize subsystem error reporting functions, and view controller and subsystem failure information.
HSUTIL	Use the HSUTIL utility to format and upload microcode to the subsystem’s storage devices.
VTDPY	Use the video terminal display (VTDPY) utility to view the controller's current configuration and performance data. It displays such parameters as processor utilization, host port activity and status, device state, logical unit state, and cache and I/O performance.
DILX	Use the disk in-line exerciser (DILX) utility to test and exercise the storage devices in the storage subsystem.

Descriptions of how to use each local utility program appear throughout the manual.

2.2 Accessing the CLI

To access the CLI, you need to connect a maintenance terminal or a computer running a terminal emulator program to the controller.

2.2.1 Connecting a Maintenance Terminal to the Controller

To connect a maintenance terminal or a computer running a terminal emulator program to the controller, use the serial part kit that came with your subsystem cabinet to makeup a serial line. The subsystem cabinet requires a nine-pin female connector. Check your host system or maintenance terminal to determine what kind of connector you need on it.

You connect the nine–pin female connector to the terminal connector on the backside of the subsystem cabinet/controller. Three nine-pin connectors appear on the back of the subsystem cabinet/controller. Two connectors exist side-by side. The terminal connector is the one situated by itself.

Connect the other end of the serial line to the maintenance terminal or computer com port.

2.2.2 Getting to a CLI Prompt

If you connected to a maintenance terminal, press enter and the CLI prompt, `HSZ20>`, appears.

To get to a CLI prompt from a terminal emulation program, follow these steps:

1. Start the terminal emulator program.
2. Set the terminal emulator program to use the serial port (com1, com2, so forth.) that you connected to the RAID Array 310 controller.
3. Set the other communications parameters to:
   - 9600 baud
   - 8 bits
   - 1 stop bit
   - No parity
   - Xon/Xoff
4. Disable **Line Wrap**, if your terminal emulator supports it.
5. Issue a connect command, if necessary, and press the Return key, or just press the Return key. Once the controller initializes, if it hasn't initialized already, the CLI prompt, `HSZ20>`, appears.

NOTE

This guide uses the designations `CLI>` and `HSZ20>` interchangeably to identify the CLI prompt.

2.2.3 Guidelines for Using the CLI

Be aware of the following when using the CLI:

The terminal Backspace key on some terminals may not act normally with the CLI. If you make a mistake in typing in a command, you may be able to make corrections using the Delete key as a Backspace key.

You do not need to enter all configuration parameters on one line. You can enter parameters by using multiple SET commands.

You need to enter only enough of each command to make the command unique (usually three characters). For example, SHO for SHOW, or SHOW THIS for SHOW THIS_CONTROLLER

The firmware waits approximately one minute after a controller restart before presenting a CLI prompt or accepting commands. This delay does not affect unit availability to the host. The delay provides time for controller internal configuration operations to finish before user commands change the configuration.

2.2.4 Entering CLI Commands

CLI commands let you control various aspects of the operation of the controller, devices, storagesets, and logical units.

NOTE

A unit experiencing heavy I/O load may be slow to respond to CLI commands.

To execute any command, you specify the command at the CLI prompt using the following basic structure:

- Command name
- Parameter
- Qualifier (if any)

For example, to increase the speed of your maintenance terminal connection, you enter the following command line:

`HSZ20>`**`set this_controller terminal_speed=19200`**

where set = a CLI command
this_controller = the part of the subsystem which you want to adjust
terminal_speed = 19200 = a parameter for the controller and its new setting

After you enter a command, either the controller displays a message indicating the result of the command, or you enter another command, such as a show command, to view the result.

2.2.5 Getting Help

The CLI provides help by displaying a list of the commands available through the CLI or more detailed information about using a specific command.

> **NOTE**
>
> Although the firmware onboard the RAID Array 310 controller uses terminology related to two-controller operation such as – failover, set other controller, and dual redundancy – the RAID Array 310 does not support two controller, dual-redundant operation.

At the CLI prompt, type: **`?`**
to view a list of the commands available through the CLI

At the CLI prompt, type a command name (or a partial command name), a space, and then a question mark to get information about using a specific command.
For example, type: **`set ?`**
The controller displays information similar to the following:

```
Your options are:
   FAILOVER
   NOFAILOVER
   OTHER_CONTROLLER
   THIS_CONTROLLER
   Unit number or container name
```

Go on to Chapter 3 to Configure the subsystem.

3

Configuring the Subsystem

This chapter describes the major steps required to configure the RAID Array 310 subsystem and also the types of storagesets that you can create.

You already began to plan your subsystem when you selected a RAID Array 310 with space available for seven devices. Configuring the Subsystem requires the following major steps:

- Setting the controller SCSI ID (Section 3.1)
- Adding devices to the controller (Section 3.2)
- Selecting a storage configuration (Section 3.3)
  - Creating storagesets (Sections 3.4 through 3.6)
  - Initializing storagesets (Sections 3.5.6 and 3.6.7)
- Saving the configuration (Section 3.6.7.3)
- Adding devices and storagesets to the host (Section 3.7)

Use the onboard controller firmware command line interface (CLI) to configure the RAID Array 310.

To access the CLI:

From a maintenance terminal, press enter and the CLI prompt, `HSZ20>`, appears.

To get to a CLI prompt from a terminal emulation program:

1. Start the terminal emulator program.
2. Set the terminal emulator program to use the serial port (com1, com2, so forth.) that you connected to the RAID Array 310 controller.
3. Set the other communications parameters to:
   - 9600 baud
   - 8 bits
   - 1 stop bit
   - No parity
   - Xon/Xoff
4. Disable **Line Wrap**, if your terminal emulator supports it.
5. Issue a connect command, if necessary, and press the Return key, or just press the Return key. Once the controller initializes, if it hasn't initialized already, the CLI prompt, `HSZ20>`, appears.

For more information on using the CLI see Chapter 2.

3.1 Setting the Controller Target SCSI IDs

Prior to setting the controller target SCSI IDs, determine how many devices or storagesets you plan to use in your RAID Array 310 subsystem. Then, specify a separate target SCSI ID for each of the first four device or storageset units you plan to configure. You can specify up to four target SCSI IDs for the controller. Valid target SCSI IDs include 0 – 7.

For example, if you plan to configure three device or storageset units, specify target SCSI IDs 0 through 2. If you plan to configure five device or storageset units, specify target SCSI IDs 0 through 3.

You can then distribute up to four device or storageset units on unique target SCSI IDs. You distribute units across target SCSI IDs when you add the units. The target SCSI ID is the first digit of the unit number.

> **NOTE**
>
> To change the SCSI ID of a RAID 310 Subsystem controller for a subsystem with an existing configuration, you must first delete the configuration.

To set the controller SCSI ID, follow these steps:

1. Access the CLI.
   Type: `HSZ20>` **`set this_controller id=0,1`**
   where set = a CLI command
   this_controller = the subsystem component with which you want to work
   id = the target SCSI Id, the parameter you want to set
   0,1 = the new settings for the parameter, target SCSI Ids 0 and 1
2. In addition to setting the controller SCSI ID, you can change any of the other controller parameters from their defaults as described in Table 3–1. Use command syntax similar to that shown in Step 1, replacing `id=0` with the name of the parameter you want to change and its setting, for example `prompt=system1`.

> **NOTE**
>
> You must set each parameter using a separate command.

> **WARNING**
>
> CACHE_POLICY=B can result in the loss of data in the event of a power failure if battery conditions are not sufficient to retain the data in the cache until power is restored. Normal battery conditions allow the cache to retain data for a minimum of 100 hours.

4. After setting the controller SCSI ID and other parameters,
   type: `HSZ20>`**`restart this_controller`**
5. To verify the SCSI ID and parameters were set,
   type: `HSZ20>`**`show this_controller full`**

Table 3–1 Controller Parameters

Parameter	Options	Default
CACHE_FLUSH_TIMER Sets the amount of time, in seconds, between flushes of the entire contents of the write-back cache to disk.	1–65535 seconds	10 seconds (Ensures cash flushed and written to storage devices on regular basis.)
CACHE_POLICY Sets the policy for the availability of RAIDsets and mirrorsets when the battery condition is low during controller initialization. A = When the controller initializes and the battery condition is low, the controller makes inoperative any RAIDset or mirrorset. B= When the controller initializes and the battery condition is low, the controller accesses any RAIDset or mirrorset in write-through mode.	CACHE_POLICY=A CACHE_POLICY=B	CACHE_POLICY=A
HOST_FUNCTION Sets the controller to provide some responses, and operational and error behaviors consistent with the specified host type.	A = Most Hosts B = IBM/RS 6000 Hosts C = Siemens Nixdorf Hosts D = Hosts running Windows NT	A
ID Sets the SCSI ID of the controller on the host SCSI bus. Set a unique SCSI ID for the controller.	0–7	None
PROMPT Allows you to specify a prompt for the controller’s firmware command line interface.	1–16 ASCII printable characters	HSZ20
TERMINAL_PARITY Sets the terminal parity to odd, even or none. No terminal parity causes the controller to not check for or transmit any parity on the terminal lines.	ODD EVEN NOTERMINAL_ PARITY	NOTERMINAL_PARITY

Table 3–1 Controller Parameters (cont.)

Parameter	Options	Default
TERMINAL_SPEED Sets the terminal speed to the specified baud rate.	300, 600, 1200, 2400, 4800, 9600, 19200	9600
TIME Sets the current data and time.	dd-mmm-yyyy:hh:mm:ss = day-month-year:hour: minute:second	None

When you finish setting the controller's SCSI ID, go on to add devices to the controller.

3.2 Adding Devices to the Controller

After you add devices physically to the subsystem cabinet, tell the controller what devices exist by adding the devices through the onboard controller firmware. Add devices either automatically or manually. Table 3–2 describes when to use each of these methods.

Table 3–2 Methods of Adding Devices

Adding Method	When To Use
Automatically	To set all devices as notransportable. The transportable/notransportable setting specifies whether you can move devices from your RAID Array 310 to another subsystem. Digital does not recommend moving devices from your RAID Array 310 and thus setting your devices notransportable. See Section 3.2.1 for more information about transportability. To allow the utility to name the devices.
Manually	To set one or more devices as transportable.

3.2.1 Understanding Transportability

Transportability specifies whether you can remove a device from a RAID Array 310 subsystem and use it in another subsystem (transportable), or not (notransportable). The difference between the two types of devices involves whether the device contains metadata or not.

The onboard controller firmware stores metadata, which improves data reliability and error detection and recovery, on a small amount of the space available on a notransportable device, but does not store this metadata on a transportable device. Options for setting transportability include the following:

- Transportable - Add a device transportable so that the controller firmware does not add metadata to the device and you can then remove the device from the subsystem to use it in another subsystem.

NOTE

You cannot use transportable devices in storagesets.

Since the controller does not add metadata to a transportable device, you cannot use a transportable device in a storageset. Therefore, only set a device transportable if you plan to use it as an individual device.

NOTE

Digital recommends making devices notransportable unless you have no other means of moving data you store on a disk in your RAID Array 310 subsystem to another subsystem.

Notransportable **-** Add a device notransportable so that the controller firmware adds metadata to the device and then you can use the device in a storageset. You cannot remove a notransportable device from the subsystem to use in another subsystem. The onboard controller firmware makes devices notransportable by default.

3.2.2 Adding Devices Automatically

The RAID Array 310 onboard controller firmware offers you two utilities to use for automatically adding devices. Table 3–3 describes when to use each method.

Table 3–3 Methods of Automatically Adding Devices

Utility	When To Use
CONFIG	To set all devices as notransportable. The transportable/notransportable setting specifies whether you can move devices from your RAID Array 310 to another subsystem. Digital does not recommend moving devices from your RAID Array. 310. See Section 3.2.1 for more information about transportability. To allow the CLI to name the devices.
CFMENU	To set one or more devices as transportable.

3.2.2.1 Adding Devices with the CONFIG Utility

The CONFIG utility automatically locates devices in your storage cabinet and names them so that the controller can recognize them.

Use the CONFIG Utility to add all devices in your storage cabinet as notransportable.

Digital recommends making your devices notransportable unless you have no other means of moving data you plan to store on a disk in your RAID Array 310 subsystem to another subsystem. For more information about transportability see Section 3.2.1

To add devices with the CONFIG Utility, follow these steps:

1. Type: `HSZ20>` **`run config`**

   The utility displays messages similar to the following depending upon how many devices you installed in your storage cabinet.

   ```
   Config Local Program Invoked
   Config is building its tables and determining what devices
   exist on the subsystem. Please be patient.
   add disk DISK100  1 0 0
   add disk DISK110  1 1 0
   add disk DISK120  1 2 0
   add disk DISK130  1 3 0
   add disk DISK200  2 0 0
   add disk DISK210  2 1 0
   add disk DISK220  2 2 0
   config - Normal Termination
   HSZ20>
   ```

2. To verify the devices added by the CONFIG Utility,

   Type: `HSZ20>` **`show devices`**

   The utility displays a message similar to the following:

   ```
   Name       Type   Port Targ  LUN          Used by
   --------------------------------------------------
   DISK100    disk      1    0    0
   DISK110    disk      1    1    0
   DISK120    disk      1    2    0
   DISK130    disk      1    3    0
   DISK200    disk      2    0    0
   DISK210    disk      2    1    0
   DISK220    disk      2    2    0
   ```

Once you finish adding devices, go to Sections 3.3 and 3.4 to select and create your storage configuration.

3.2.2.2 Using CFMENU

The CFMENU utility adds devices as notransportable by default. However, using the CFMENU utility you can also add devices as transportable.

Digital recommends making devices notransportable unless you have no other means of moving data stored on a disk in your RAID Array 310 subsystem to another subsystem. For more information about transportability see Section 3.2.1

Using the CFMENU utility you can add all new devices, or specify each device you want to add.

To invoke the CFMENU Utility, type the following:

`HSZ20>` **`run cfmenu`**

The main menu appears as shown in Figure 3–9.

3.2.2.2.1 Adding Devices as Notransportable Using CFMENU

To add all new devices as notransportable, follow these steps:

1. From the main menu, type: `Enter menu choice (1,8) [8] ?` **`1`** and press the Return key.

   The device menu appears similar to the screen show in Figure 3–1.

Figure 3–1 The Device Menu

```
-----------------------------------CFMENU Configuration Menu Utility------------------------
                                   |Unconfig'd|Config'd Device  Product  Stor.set  Stor.set  Chnk  Trn-  In-   Re-       WW
DEVICE MENU:                       | Dev.PTLs | PTLs    Name     ID       Name      Typ/Sz    Size  sp.   it'd  duc  Unit PB
1. Add a device from list of PTLs  |^100 (dsk)|
   not configured (marked with ^)  |^110 (dsk)|
2. Delete an unbounded device      |^120 (dsk)|
   (marked with *)                 |^130 (dsk)|
3. Add all devices from list of    |^200 (dsk)|
   PTLs not configured (marked     |^210 (dsk)|
   with ^)                         |^220 (dsk)|
4. Delete all unbounded devices    |          |
   (marked with *)                 |          |
5. Return to main menu             |          |
```

2. At the Device Menu prompt: `Enter menu choice (1,8) [8] ?`

   Type **`3`** and press the Return key.

3. At the prompt: `Disks will all be set NOTRANSPORTABLE Continue (y/n) [y] ?`
4. Type **`y`** and press the Return key.
5. The utility confirms each device that it adds. All added devices appear in the Config'd PTLs column.
6. At the Device Menu prompt: `Enter menu choice (1,5) [5] ?`
   Type **`5`** and press the Return key.
   The Main menu appears.

Once you finish adding devices, go to Sections 3.3 and 3.4 to select and create your storage configuration.

3.2.2.2.2 Adding Devices as Transportable or Notransportable Using CFMENU

To add one new device at a time, follow these steps:

1. At the main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **`1`** and press the Return key.
2. The Device menu appears similar to the screen show in Figure 3–9.
3. At the Device Menu prompt: `Enter menu choice (1,8) [8] ?`
   Type: **`1`** and press the Return key.
4. The utility displays a prompt similar to the following:
   `Add disk device at PTL 100 (y/n/q) [n] ?`
5. At the prompt: `Add disk device at PTL 100 (y/n/q) [n]`
   Type: **`y`** and press the Return key.
6. At the prompt: `Set device NOTRANSPORTABLE (y/n) [y] ?`
   Type: **`y`** or **`N`** and press the Return key.
   The disk appears in the Config'd PTLs column and the utility prompts you to add the next device, if any.
7. At the Device menu prompt, `Enter menu choice (1,5) [5] ?`

Type: **`5`** and press the Return key.

The Main menu appears.

After you add the devices, if you plan to use any of the devices individually and you added them as notransportable, go to Section 3.5.6 to initialize them. Otherwise go to Sections 3.3 and 3.4 to select and create your storage configuration.

3.2.3 Adding Devices Manually

After you add devices physically to the subsystem cabinet, tell the controller what devices exist by adding the devices through the onboard controller firmware. The process of adding devices manually requires you to:

1. Determine device names
2. Name the devices

3.2.3.1 Determining Device Names

Use the following format to name your devices:

DISKptl

where: Disk = the type of device
ptl = p = the device's port address on the controller
t = the device's target address on the port
l = the device's LUN number on the target

Determine the ptl for each of your devices as follows:

Port = the channel on the RAID 310 controller to which this device is connected. The RAID 310 controller has two channels. See the RAID Array 310 Deskside Subsystem User's Guide EK–SMCPL–UG to determine which cabinet slots are on each channel. If you do not have a full compliment of devices in your RAID Array 310 cabinet, install your devices in cabinet slots so that they are divided between the two channels to maximize system performance.

Target = the device's SCSI address. This is determined by the cabinet slot in which you installed the device. See the RAID Array 310 Deskside Subsystem User's Guide EK–SMCPL–UG to determine the SCSI address of each slot.

LUN = the LUN number for each device installed in a cabinet slot is 0.

3.2.3.2 Naming Devices

Use the following CLI command syntax to manually add/name each device:

For example, type: **`Add disk disk100 1 0 0`**

where: add = a CLI command
disk = the device type
disk100 = the device name, made up of the device type and the ptl you determined for the device.
1 0 0 = the ptl. Type the ptl with a space between each number.

After you add the devices, if you plan to use any of the devices individually and you added them as notransportable, go to Section 3.6.7 to initialize them. Otherwise go to Sections 3.3 and 3.4 to select and create your storage configuration.

3.3 Selecting a Storage Configuration

The storage configuration options available to you depend upon your storage needs and the number of disks that you purchased for your RAID Array 310 storage cabinet. Table 3–4 describes the storage options and the minimum number of disks required to implement each.

You can use multiple types of storagesets within a single subsystem, if you have the disk device resources to support them.

Table 3–4 Configuration Options

Storage Method	Type of Storageset	Number of Devices Required	Benefit
Individual Devices	Disk Drive	1 - 7	Provides the storage capacity and access speed of the disk used.
RAID 0	Stripeset	2 - 7 devices	Provides high I/O performance for applications requiring either high I/O request rates or high data transfer rates.
RAID 1	Mirrorset	2 - 3 devices per mirror set, up to 3 mirror sets per RAID Array 310	Provides maximum availability, protecting data against disk failure by replicating all data stored.
RAID 0 + 1	Striped mirrorsets	2 - 3 mirror sets	Provides high I/O performance and maximum availability.
RAID 3/5 A redundant-stripeset combining the optimized data transfers of RAID 3 with the striping of parity of RAID 5.	RAIDset	3 - 7 devices	Provides affordable data availability and redundancy.

The following sections provide detailed descriptions of each of the storage methods/types of storagesets.

3.3.1 Individual Devices

You can use individual disk drives in your RAID Array 310 subsystem. However, using the RAID Array 310 solely for individual disks will not achieve any substantial performance benefits over conventional subsystem storage options.

3.3.2 RAID Options

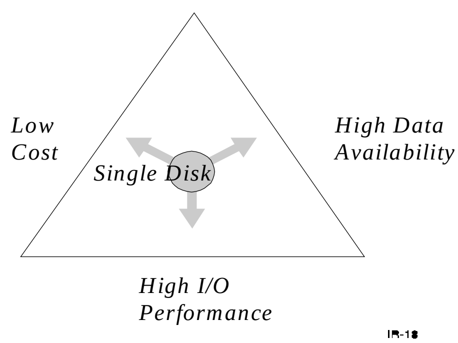
RAID means Redundant Array of Independent Disks. It is a way of configuring multiple disk drives to achieve high data availability. RAID is implemented as a set of multiple disks, called an array, and a specialized array controller, which manages the distribution of data across the disk drives.

A RAID array, whether it contains two, five or seven drives-, looks like one or more large disk drives to the user. You use a RAID array just as you would any other drive. You can partition it if you want, and you do not need to make any application changes to realize the benefits of RAID. A RAID array can provide higher levels of data availability and performance than a single disk drive of similar capacity.

Data for a given file is divided into chunks that can be written across multiple drives. A **chunk** is a group of contiguous data blocks which are stored on a disk drive. By using more than one drive, the array provides higher data transfer rates when compared to a single large drive. Depending on the RAID level used, arrays can also provide redundancy to protect the data availability. Arrays provide redundancy in two main ways: by mirroring and by generating parity.

The requirements of all system users fall into the triangle shown in Figure 3–2.

Figure 3–2 Graphical Analysis of RAID Solutions



For some users, a system has three important attributes: low cost, high performance, and low data availability. These users occupy the left leg of the triangle. Other users may lose millions of dollars in business if their system is down for only a short time. For them, additional dollars spent to provide high data availability is dollars well spent. They occupy the right leg of the triangle.

For still others, getting the job done quickly outweighs any other attribute. They occupy the bottom of the triangle.

Systems that use non–RAID (standard single disk) storage fall in the center of the triangle, equidistant from the three attributes (low cost, high data availability, and high I/O performance). To determine the appropriate storage method, you can rank the importance of the attributes you desire and place them in a general area within the triangle.

The range of RAID levels available in the RAID Array 310 provides a solution for most users. RAID is not a single, discrete product. RAID comes in several varieties, or levels. Each RAID level offers tradeoffs among availability, performance, and cost. The levels are numbered 0 through 5/3 (for example, RAID 1, RAID 5/3, and so on). These numbers are not a quality or performance ranking. They simply distinguish various RAID configurations. The StorageWorks RAID Array 310 Subsystem supports the following types: RAID 0 (striping), RAID 1 (mirroring), RAID 0+1 (mirroring and striping), and RAID 3 and 5 (striping data and striping parity). For more information about the different RAID levels, see the following Sections, or *The Digital Guide To RAID Storage Technology*, Order No. EC-B1960-45.

3.3.3 RAID 0 / Stripeset

RAID level 0 is known as striping. Striping spreads data across multiple disks, breaking the user data into segments designated as "chunks." In a four disk stripeset, the first chunk is written on disk 1, the second on disk 2, the third on disk 3, the fourth on disk 4, the fifth on disk 1, and so on.

The system administrator sets the chunksize based upon application requirements. If the chunksize is set to be relatively large related to the average input/output (I/O) size, all of the disks may be able to execute different read/write requests simultaneously. If there are large numbers of frequently accessed files, this may be especially beneficial.

If the chunksize is set significantly smaller than the average I/O size, then most or all of the disks in the stripeset will be able to transfer data for a single request in parallel. This method increases the data transfer rate for large I/Os.

RAID level 0 provides high performance for a wide variety of I/O intensive applications. Depending on the hardware configuration and the chunksize set, RAID level 0 improves either data transfer rate or I/O request rate.

CAUTION

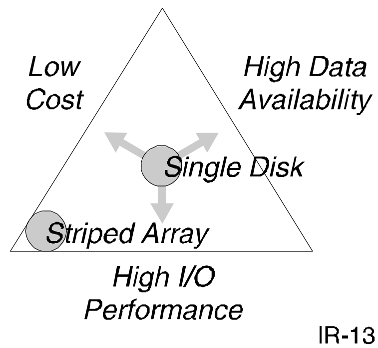
If any member of a RAID level 0 stripeset fails, all data is lost from the entire set.

Figure 3–3 depicts a graphical analysis of RAID level 0. For I/O applications RAID 0 is suited to the following:

- Storing program image libraries or run–time libraries for rapid loading
- Storing large tables or other structures of read–only data for rapid application access
- Collecting data from external sources at a very high data transfer rate

Availability of data is proportionate to the number of drives; worse than a single drive. In terms of cost of storage, the user only purchases the capacity needed to store his data.

Figure 3–3 RAID 0 – Data Availability, Performance, and Cost Analysis



In analyzing the triangle shown in Figure 3–3, RAID 0 (Striped Array) has a high input/output performance (Striped Array symbol is closest to the High Input/Output side of triangle). RAID 0 is low as to High Data Availability (Striped Array symbol located farthest from High Data Availability side of triangle). The cost of storage is low (Striped Array symbol located next to Low Cost side of triangle) as shown.

3.3.4 RAID 1 / Mirrorset

RAID level 1 (disk mirroring) protects data against disk failure by replicating all data on each member of the mirrorset. RAID level 1 offers extremely high data reliability, albeit at a relatively high cost (because all disks are duplicated). For some I/O intensive applications, a RAID level 1 array can improve performance significantly over a single disk.

The controller's Mirroring Option includes the following features:

- Real-time maintenance of up to six identical copies of data on mirrorsets of separate disks or storagesets attached to a single controller.
- Striping of mirrorsets, for high-performance access to large amounts of highly available data. This feature allows for the addition of mirroring to disks that are currently stripeset members (for users with existing stripesets who wish to use mirroring technology).
- Automatic replacement of a failed member of a mirrorset with a spare disk, if a suitable (sufficiently large) spare disk has been designated. As with the RAID level 5 (RAIDsets) option, either a best-fit or a best-performance replacement policy may be designated.
- Inclusion of multiple disk types in a single mirrorset (the capacity of the mirrorset unit is bounded by that of the smallest disk in the mirrorset).
- Ability to increase or decrease the number of members in a mirrorset as requirements change.
- Flexible policy options for determining both how read requests are satisfied and the speed of copying when a new member is being added.

- In addition the mirroring option supports a feature which allows a system administrator to create an identical copy of any controller disk or stripeset unit, and then dissociate it from the original. This feature is called "cloning." Many users will find this feature helpful for doing back up operations of an application's data by:
  - Making clones of its disk or stripeset unit momentarily quiescing the application
  - Dissociating the clones from the primary units
  - Reenabling the application with only the primary units

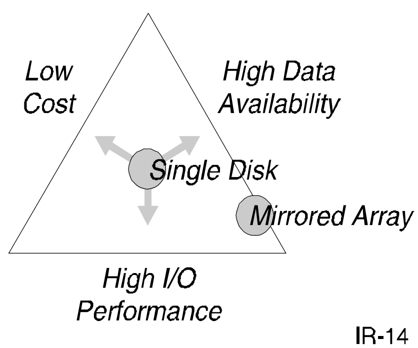
Like the RAID level 5 option, write performance can be enhanced by enabling write-back caching of user data for each mirrorset.

Figure 3–4 shows a graphical analysis of the data availability, performance, and cost for RAID 1. RAID 1 performs well for a variety of mission-critical applications such as system disks, master files, database journals, etc.

Data availability is excellent. If one disk fails, there is no loss of data availability. Loss of data availability occurs only when the second disk of a pair fails before the first can be replaced.

In terms of cost of storage, RAID 1 is expensive. The user must purchase dual disk drives. One drive is used for data storage, the other for data redundancy. The expense increases for greater than two member mirrorsets.

Figure 3–4 RAID 1 – Data Availability, Performance, and Cost Analysis

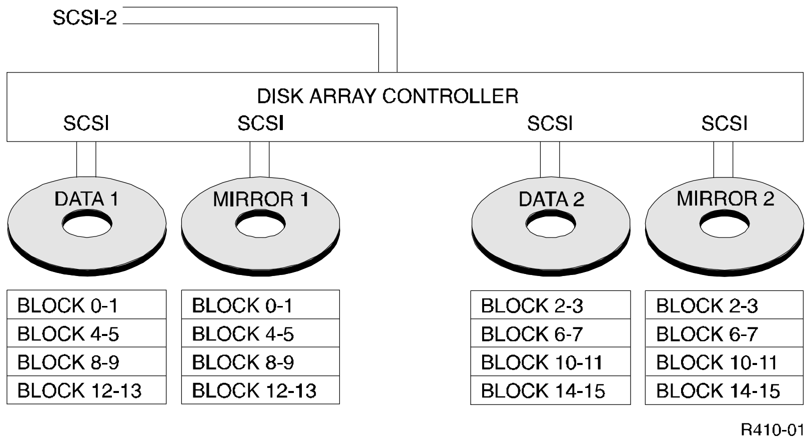


In analyzing the above triangle, RAID 1 (Mirrored Array) has high data availability (Mirrored Array symbol is close to the High Availability side). A RAID 1 Array has High Input/Output performance as shown. The cost factor is relatively high (Mirrored Array symbol is far from Low Cost side of triangle).

3.3.5 RAID 0 + 1 / Striped Mirrorset

RAID level 0 + 1, which is described as stripesets whose members are mirrorsets, are called striped mirrorsets. RAID 0+1 is the combination of striping and mirroring, implemented by striping data across mirrored sets as shown in Figure 3–5. It provides the best performance of any type of RAID by combining the performance advantages of RAID 0 and RAID 1.

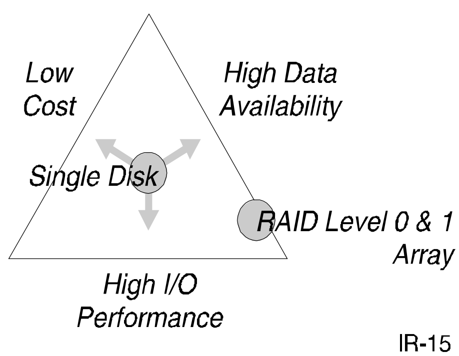
Figure 3–5 Diagram of RAID 0+1: Transparent Mirroring



RAID 0+1 provides data redundancy and is beneficial for any critical response-time application. It also provides disaster tolerance. If a drive fails in a RAID 0+1 array, you can continue to use the array normally since data from its mirrored drive is retrieved. It is unique because more than one drive can fail without data loss as long as at least one drive in each mirror set survives.

Figure 3–6 presents a graphical analysis of RAID 0+1. For I/O applications, this level is suited where response time is critical. Speed is based on chunk–size/request-size ratio. You can optimize for request rate or data rate. As for cost, RAID 0+1 uses no more physical disks than mirroring alone as in a RAID 1 configuration. The incremental cost of RAID 0+1 is that of striping through the controller. This makes RAID 0+1 an excellent bargain for any data whose value justifies placing it on mirrored storage.

Figure 3–6 RAID 0+1 – Data Availability, Performance, and Cost Analysis



In analyzing the above triangle, a RAID 0+1 Array has high data availability (RAID 0+1 Array symbol is closest to the High Data Availability side) and a High Input/Output performance. The cost factor is relatively high (RAID 0+1 Array symbol is far from low cost side of triangle).

3.3.6 RAID 3 / RAIDset

Industry standard RAID level 3 achieves higher bandwidths as a result of transferring a part of each I/O's data from each RAIDset member in parallel. To achieve high bandwidths with conventional fixed-block disks (typically 512 data bytes in size), all I/O requests must specify an amount of data equal to the member block size, multiplied by the number of members in the RAIDset, minus one. Also, the requests' starting addresses must be aligned so that correspondingly located data from each member is transferred. To permit this data transfer to take place in parallel, industry standard RAID level 3 often requires special disks or configurations to ensure that all disks in the RAIDset are rotating in perfect synchronization.

Industry standard RAID level 3 performs as though the RAID level 3 RAIDset is a single disk with a specific large (virtual) sector size. This results in substantial performance penalties for I/Os that are not perfectly aligned multiples of the larger data byte size. Few applications use extremely large I/O sizes (and these may not easily be modified to use a multiple of the RAID level 3 virtual sector size). In any event, many operating systems can not easily accommodate virtual disks with unconventional sector sizes.

Digital's implementation of RAID level 3 for the RAID Array 310 achieves higher bandwidth levels without the virtual sector size or special device/configuration disadvantages. This is achieved with special algorithms related to RAID level 5 technology, but without the write performance penalty associated with conventional RAID level 5 (but not with conventional RAID level 3) implementation.

For convenience, this capability is controlled by setting the RAIDset's chunk size to a lower value, and performing sequential write operations (using the write-back cache mode). This permits higher bandwidth performance results approaching industry standard RAID level 3 operation. With the capability of setting chunk size, you can conveniently choose between more bandwidth-oriented or more throughput-oriented performance using the same configuration and CLI commands. When you specify intermediate chunk sizes, you realize large I/O benefits from RAID level 3 technology, while getting smaller I/O benefits from RAID level 5 technology.

NOTE

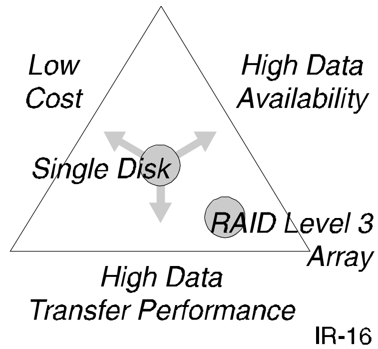
Chunk size is set with the CLI INITIALIZE CHUNKSIZE= command. The initialize command also erases any data existing on the device.

Figure 3–7 illustrates the cost, data availability, and performance of RAID 3. For I/O applications, this level is advantageous for:

- High-volume data collection, such as seismic or telemetric and video-on-demand
- Processing of large images

Data availability is excellent and in terms of cost of storage, RAID 3 is good. The user needs to purchase just one additional disk to provide the redundancy.

Figure 3–7 RAID 3 – Data Availability, Performance, and Cost Analysis



In analyzing the above triangle, a RAID 3 array has good data availability (RAID 3 Array symbol is near the High Data Availability side). RAID 3 is considered good as to High Input/Output performance. The cost factor is relatively high (RAID 3 Array symbol is far from low cost side of triangle).

3.3.7 RAID 5 / RAIDset

RAID level 5 stripes data and rotates parity across all disks in the RAIDset. The controller combines incoming data with existing parity data.

RAID level 5 is suited for applications whose I/O loads consist predominantly of a large number of asynchronous read requests. Transaction processing and office automation applications often fall into this category. It also is good for data transfer intensive applications, such as image analysis, which make mostly read requests. It is not as well suited for write intensive applications (such as data entry, scientific or engineering data collection).

NOTE

If using RAID level 5, all data in the RAIDset will be lost if a second drive fails in the same set before the first failed drive is replaced.

The RAID facility available as an optional feature in conjunction with the write-back cache module, combines elements of RAID level 5 and RAID level 3 technology to provide affordable data availability (compared to RAID Level 0) without the I/O performance penalties normally associated with RAID level 5.

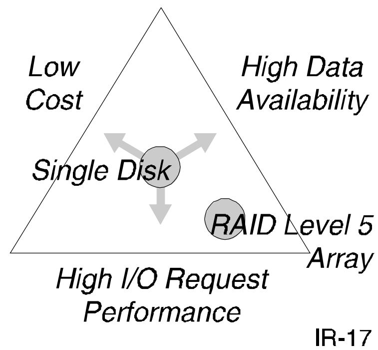
In the RAID Array 310, the RAID level 5 facility uses a distributed data mapping technique just like that used for disk striping. A powerful measure of protection against hardware component failure is added by reserving some of the blocks in each RAIDset's disk devices for the storage of redundant information. This redundant information allows the contents of any block of application data stored in the RAIDset, to be *regenerated* in the case of a disk device failure on which the data is stored (as long as the remaining RAIDset members are functioning properly).

RAID levels 3 and 5 are sometimes called parity RAID levels, because the redundant information they store is in the form of a parity block which corresponds to data blocks in each of the RAIDset's disk devices. *Parity* is any kind of checksum that allows the regeneration of unretrievable data. Parity is typically combined with data stored in positionally corresponding blocks of other disk devices in the RAIDset to regenerate the missing data.

For detailed information about RAID technology, refer to *The RAIDBOOK—A Source for RAID Technology*.

The triangle shown in Figure 3–8 analyzes RAID level 5. For I/O applications, interactive transaction processing, multi-user file services, and office environment applications, data availability is excellent. In terms of cost of storage, RAID 5 is good. The user needs to purchase just one additional disk to provide redundancy.

Figure 3–8 RAID 5 – Data Availability, Performance, and Cost Analysis



In analyzing the above triangle, a RAID 5 mirrored array has a high data availability (RAID 5 Array symbol is near the High Data Availability side). RAID 5 is considered good as to High Input/Output performance. The cost factor is relatively high (RAID 5 Array symbol is far from low cost side of triangle).

3.4 Creating Storagesets

Once you select the type of storagesets that you want to use in your subsystem, you must create them. The process of creating storagesets requires the following major steps:

- Configuring devices into storagesets to meet the performance and availability requirements of your subsystem
- Initializing the storagesets

3.4.1 Selecting a Configuration Method

The CLI offers you the following methods by which you can create a storageset

- Automatically
  –Use CFMENU to automatically create storagesets
- Manually

–Use CLI commands to manually create storagesets

Table 3–5 describes when to use each of these methods.

Table 3–5 Configuration Methods

Configuration Method	When To Use
Automatically	To create any configuration You must use the default chunksize (256 Blocks) for RAIDsets and stripesets when you configure automatically. For more information about chunksize, see Table 3–6, or Section 3.6.7.2.
Manually	To specify a chunksize. To change the default chunksize for a stripeset or RAIDset, configure manually.

Sections 3.5 and 3.6 describe each of the steps required to configure your subsystem using the automatic or manual methods.

3.4.2 Accessing the CLI

Use the controller firmware command line interface (CLI) to configure your RAID Array 310. See the introduction to this chapter to access the CLI. See Chapter 2 for information on the CLI.

3.5 Configuring Automatically

To configure your subsystem automatically use the CFMENU. The CFMENU allows you to automatically:

- Create storagesets
- Create a spareset
- Initialize storagesets
- Add devices and storagesets to the host

3.5.1 Invoking the CFMENU Utility

To invoke the CFMENU Utility, type the following:

`HSZ20>` **`run cfmenu`**

The main menu appears as shown in Figure 3–9.

Figure 3–9 The CFMENU Main Menu

```
------------------------------------CFMENU Configuration Menu Utility------------------------
                                   |Unconfig'd| Config'd   Device     Product     Stor.set Name  Stor.set  Chnk  Trn-  In-   Re-        WPWB
           MAIN MENU:              | Dev.PTLs |  PTLs      Name         ID                       Typ/Sz   Size  sp.   it'd  duc   Unit
1. Add/delete devices              |^100 (dsk)|
2. Add/delete mirrorsets           |^110 (dsk)|
3. Add/delete stripesets           |^120 (dsk)|
4. Add/delete raidsets/sparesets/  |^130 (dsk)|
   failedsets                      |^200 (dsk)|
5. Partition processing            |^210 (dsk)|
6. Initialize devices and/or       |^220 (dsk)|
   storagesets                     |          |
7. Add/delete units                |          |
8. Setup terminal                  |          |
9. Exit CFMENU                     |          |
```

Table 3–6 describes the information the CFMENU displays.

Table 3–6 The CFMENU Display

CFMENU Display Heading	Definition	Description
Main Menu		The options you can select from this screen.
Unconfig’d PTLs	Unconfigured PTLs	Devices installed in the subsystem cabinet that have not yet been made known to the controller.
Config’d PTLs	Configured PTLs	Devices that have been made known to the controller. (Devices are made known to the controller when you add/name them.)
Device Name		The name CFMENU automatically assigns to each device. The name contains both the device type and PTL. To change the device name that CFMENU assigns (not recommended), you must delete the device and add it manually. See Section 3.4.1 Adding Devices Manually.
Product ID		Information identifying the device model.

Table 3–6 The CFMENU Display (cont.)

CFMENU Display Heading	Definition	Description
Stor.set Name	Storageset name	The name of a configured storageset, for example S1. CFMENU assigns this name when you create a storageset. CFMENU uses the following naming conventions: a letter to identify the type of storageset and a number to make each storageset of a type unique. S# = Stripeset and Striped Mirrorset, M#= Mirrorset, R#= RAIDset, To change the storageset name that CFMENU assigns, you must delete the device and add it manually. See Section 3.6 Configuring Manually.
Stor.set Typ/Sz	Storageset type and size	A code to identify the type of storageset and also identifies the number of members of the storageset. For example, MIR/2, indicates a mirrorset with two members. CFMENU uses the following designator for each type of storageset: STR/ = Stripeset and Striped Mirrorset, MIR/ = Mirrorset, RAD/ = RAIDset.
Chnk Size	Chunk Size	The number of blocks written to one RAIDset or stripeset member before data is written to the next member. unk appears in this column until you set the chunksize. You set the chunksize when you initialize. Do not use CFMENU to set chunksize for a Stripeset or RAIDset during initialization, incorrect chunksize may result.
Trnsp	Transportable	Identifies with a Y or N whether a device is transportable.
Init'd	Initialized	Identifies with a Y or N whether a device/storageset has been initialized. You cannot use an uninitialized storageset in a unit or an uninitialized mirrorset in a stripeset.
Reduc	Reduce	Indicates whether a RAIDset is missing a member and thus running reduced.
Unit		Displays the unit number by which the host recognizes the device or storageset.
WP	Write Protection	Identifies with a Y or N whether a unit is write protected.
WB	Write Back	Identifies with a Y or N whether a unit uses write-back caching.

3.5.2 Adding/Deleting Stripesets

You can use the CFMENU to add or delete stripesets.

3.5.2.1 Adding Stripesets

You add stripesets by selecting the devices to use in each set.

> **NOTE**
>
> To add striped mirrorsets, first create the mirrorsets, then, when you add stripesets, the CFMENU will prompt you to specify which mirrorsets to use in a stripeset.

To create stripesets, follow these steps:

1. At the Main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **`3`** and press the Return key.
   The Stripeset menu appears similar to the screen show in Figure 3–10.

Figure 3–10 The Stripeset Menu

```
------------------------------------CFMENU Configuration Menu Utility------------------------
                             |Unconfig'd| Config'd   Device      Product      Stor.set Stor.set Chnk Trn-  In-  Re-       WW
STRIPESET MENU:              | Dev.PTLs |   PTLs      Name         ID           Name    Typ/Sz  Size sp.  it'd duc Unit PB
1. Create a stripeset        |          | disks:  120 DISK120  RZ29B (C) DEC                          N     N
   (eligible entities marked |          |           ^ DISK130  RZ29B (C) DEC                          N     N
   by ^)                     |          |       130 ^ DISK210  RZ29B (C) DEC                          N     N
2. Delete an unbounded       |          |       210 ^ DISK220  RZ29B (C) DEC                          N     N
   stripeset (marked by *)   |          |       220 ^ DISK100  RZ29B (C) DEC   *   S1     STR/3   unk        N
3. Delete all unbounded      |          |strps:  100  DISK110  RZ29B (C) DEC            "       "          "
   stripesets (marked by *)  |          |        110  DISK200  RZ29B (C) DEC            "       "          "
4. Return to main menu       |          |        200
```

2. At the Stripeset menu prompt, `Enter menu choice (1,4) [4] ?`
   Type: **`1`** and press the Return key.
3. At the prompt, `Enter number of members for the Stripeset (2..14) [] ?`
   Type: **`the number of members for the stripeset`** and press the Return key.
4. At the prompt, `Include DISK100 as a member (y/n/q) [n] ?`
   Type **`Y`** and press the Return key to use this disk in the stripeset.
5. The utility prompts you to select each available device or mirrorset until you have selected the specified number. Then the utility displays a message similar to the following: `add stripeset S1  DISK100 DISK110 DISK210`
   indicating the name of the added stripeset, S1, and the devices used for that stripeset.
6. Press the return key.
   The utility now displays this stripeset and devices used for it in the *Config'd PTLs* column.
7. At the Stripeset menu prompt, `Enter menu choice (1,4) [4] ?`

   Type **`4`** and press the Return key to return to the Main menu. The CFMENU Main Menu screen displays the stripeset in the *Config'd PTLs* column.

After you create stripesets, go on to Section 3.5.6 to initialize them

3.5.2.2 Deleting Stripesets

If you decide that you do not want a stripeset that you just created, you can delete it and create a new one. If you decide to delete a stripeset on which you have stored data, backup the data before deleting the stripeset.

To delete a stripeset or sets, follow these steps:

1. At the Stripeset menu prompt, `Enter menu choice (1,7) [7] ?`
   You can type **`2`** to specify the stripeset that you want to delete, or you can type **`3`** to delete all stripesets that you just created.

   If you type **`3`** and press the Return key, the utility displays the name of each stripeset as it deletes it.

   If you Type: **`2`** and press the Return key, the utility prompts you to select one of the existing stripesets similar to the following, `Delete S1 (y/n/q) [n] ?`
2. Type **`Y`** and press the return key to select the stripeset you want to delete. The utility displays the name of the stripeset as it deletes it.
3. At the Stripeset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type **`7`** and press the Return key to return to the Main menu.

3.5.3 Adding/Deleting Mirrorsets

You can use the CFMENU to add or delete mirrorsets.

3.5.3.1 Adding Mirrorsets

You add mirrorset by selecting the devices (between two to six) for each set.

To create mirrorsets, follow these steps:

1. At the Main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **`2`** and press the Return key.
2. The Mirrorset menu appears similar to the screen show in Figure 3–11.

Figure 3–11 The Mirrorset Menu

```
----------------------------------- CFMENU Configuration Menu Utility-------------------------
                             |Unconfig'd| Config'd     Device      Product     Stor.set Stor.set Chnk Trn- In-  Re-       W W
MIRRORSET MENU:              | Dev.PTLs |   PTLs        Name          ID         Name    Typ/Sz  Size sp.  it'd duc  Unit P B
1. Create a mirrorset        |          |disks:  120 ^ DISK120  RZ29B  (C) DEC                          N    N
   (eligible devices marked  |          |        130 ^ DISK130  RZ29B (C) DEC                           N    N
   by ^)                     |          |        220 ^ DISK220  RZ29B (C) DEC                           N    N
2. Delete an unbounded       |          |miror: 100    DISK100  RZ29B (C) DEC  *   M1     MIR/2                N   N
   mirrorset (marked by *)   |          |       200    DISK200  RZ29B (C) DEC      "        "                  "   "
3. Delete an unbounded       |          |       110    DISK110  RZ29B (C) DEC  *   M2     MIR/2                N   N
   mirrorsets (marked by *)  |          |       210    DISK210  RZ29B (C) DEC      "                           "   "
4. Mirror a disk (eligible   |          |
   devices marked by +)      |          |
5. Unmirror a disk           |          |
   (eligible devices marked  |          |
   by -)                     |          |
6. Change membership of a    |          |
   mirrorset (submenu)       |          |
7. Return to main menu       |          |
```

3. At the Mirrorset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type: **`1`** and press the Return key.
4. The utility displays a prompt similar to the following:
   `Enter number of members for the mirrorset (1..6) [] ?`
   Type: **`the number of members for the mirrorset`** and press the Return key.
5. At the prompt, `Include DISK100 as a member (y/n/q) [n] ?`
   Type **`Y`** and press the Return key to use this disk in the mirrorset.
   The utility prompts you to select each available device until you have selected the specified number of devices.
6. After you select the last of the specified number of devices, the utility displays a message similar to the following: `add mirrorset M1  DISK100 DISK200`
   indicating the name of the added mirrorset, M1, and the devices used for that mirrorset. The utility now displays the devices used for this mirrorset in the *Config'd PTLs* column.
7. At the Mirrorset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type **`7`** and press the Return key to return to the Main menu. The CFMENU Main Menu screen displays the mirrorset in the *Config'd PTLs* column.

After you create Mirrorsets, go on to Section 3.5.6 to initialize them

3.5.3.2 Deleting Mirrorsets

If you decide that you do not want the mirrorset that you just created, you can delete it and create a new one. If you decide to delete a mirrorset on which you have stored data, backup the data before deleting the mirrorset.

To delete a mirrorset or sets, follow these steps:

1. At the Mirrorset menu prompt, `Enter menu choice (1,7) [7] ?`
   You can type **2** to specify the mirrorset that you want to delete, or you can type **3** to delete all mirrorsets that you just created.

   If you type **3** and press the Return key, the utility displays the name of each mirrorset as it deletes it.

   If you Type: **2** and press the Return key, the utility prompts you to select one of the existing mirrorsets similar to the following, `Delete M1 (y/n/q) [n] ?`
2. Type **Y** and press the return key to select the mirrorset you want to delete. The utility displays the name of the mirrorset as it deletes it.
3. At the Mirrorset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type **7** and press the Return key to return to the Main menu.

3.5.4 Adding/Deleting Striped Mirrorsets

To add striped mirrorsets, follow the steps specified in Section 3.5.3 on how to create mirrorsets. Then follow the steps specified in Section 3.5.2 on how to create stripesets.

After you create striped mirrorsets, go on to Section 3.5.6 to initialize them.

3.5.5 Adding/Deleting RAIDsets

You can use the CFMENU to add or delete mirrorsets.

3.5.5.1 Adding RAIDsets

You add RAIDsets by selecting the devices to use in the set (between three and seven) and by setting any of the parameters described in Table 3–7.

Table 3–7 RAIDset Parameters

Parameter	Options	Default
Policy - Defines the criteria which the controller uses to select the device to replace one that fails in a RAIDset.	Best_Performance Selects a device (from the spareset) to replace a device that fails in a RAIDset using the following criteria: 1. Achieves best performance of the mirrorset. 2. Most closely matches the sizes of the remaining devices in the RAIDset. Best_Fit Selects a device (from the spareset) to replace a device that fails in a RAIDset using the following criteria: 1. Most closely matches the sizes of the remaining devices in the RAIDset 2. Achieves best performance of the RAIDset. NoPolicy Does not select a replacement; allows the RAIDset to operate in a reduced state.	Best_Performance
Reconstruct – Specifies the rate (speed) at which the controller reconstructs a RAIDset after adding a replacement drive	Normal Balances overall performance of controller against RAIDset reconstruction operations Fast Gives priority of controller operation to reconstruction of RAIDset and may reduce controller performance to the host system. NoReconstruct	Normal
Reduced – Specifies whether a RAIDset is missing a member.	NoReduced Specifies that RAIDset being added contains the minimum number of members. Reduced Specifies that a RAIDset being add is already missing a member	NoReduced

To add RAIDsets, follow these steps:

1. At the Main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **`4`** and press the Return key.
   The RAIDset menu appears similar to the screen show in Figure 3–12.

Figure 3–12 The RAIDset Menu

```
--------------------------------------CFMENU Configuration Menu Utility-----------------------
                             |Unconfig'd|  Config'd  Device      Product      Stor.set Stor.set Chnk Trn- In-  Re-       W W
RAIDSET MENU:                | Dev.PTLs |      PTLs  Name          ID           Name    Typ/Sz  Size sp.  it'd duc  Unit P B
1. Create a raidset (eligible|          |disks: 130 DISK130  RZ29B  (C) DEC                              N    N
   devices marked by ^)      |          |         ^ DISK100  RZ29B (C) DEC  *   R1      RAD/3   unk       N    N
2. Delete an unbounded raidset          |RAID5: 100 DISK110  RZ29B (C) DEC               "       "        "    "
   (marked by *)             |          |       110 DISK200  RZ29B (C) DEC               "       "        "    "
3. Delete all unbounded      |          |       200 DISK120  RZ29B (C) DEC  *   R2      RAD/3   unk       N    N
   raidsets (marked by *)    |          |Strps  120 DISK210  RZ29B (C) DEC               "       "        "    "
4. Add/delete device in      |          |       210 DISK220  RZ29B (C) DEC               "       "        "    "
   SPARESET or FAILEDSET     |          |       220
   (submenu)                 |          |
5. Replace member of a reduced          |
   raidset                   |          |
6. Return to main menu       |          |
```

2. At the RAIDset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type: **`1`** and press the Return key.

3. At the prompt, `Enter number of members for the stripeset (3..14, or 2(only if reduced RAIDset)) [] ?`
   Type: **`the number of members for the RAIDset`** and press the Return key.

4. At the prompt, `Include DISK100 as a member (y/n/q) [n] ?`
   Type **`Y`** and press the Return key to use this disk in the RAIDset.

5. The utility prompts you to select each available device until you have selected the specified number of devices.

6. After you select the last of the specified number of devices, the utility prompts you to specify parameters for this RAIDset as follows:

7. At the prompt,
   `Enter policy for auto-replacement of failed member (0 = BEST_PERFORMANCE, 1 = BEST_FIT,2 = NOPOLICY (no auto-replacement)) [BEST_PERFORMANCE] ?`
   Type: **`number of desired replacement policy`** and press the return key.

8. At the prompt, `Enter priority for reconstruct = NORMAL, 1 = FAST, 2 = NORECONSTRUCT) [NORMAL] ?`
   `Press the Return key` to accept the default.

9. At the prompt, `Is this a previously REDUCED raidset (y/n) [n] ?`
   Type: **`Y`** or **`N`** and press the Return key.

10. The utility displays a message similar to the following: `add raidset R1 POLICY=BEST_PERFORMANCE RECONSTRUCT=NORMAL DISK100 DISK110 DISK 200`
    indicating the name of the added RAIDset, R1, the selected parameters, and the devices used for that RAIDset.

11. At the RAIDset menu prompt, `Enter menu choice (1,7) [7] ?`
    Type **`7`** and press the Return key to return to the Main menu. The CFMENU Main Menu screen displays the RAIDset in the *Config'd PTLs* column.

After you create RAIDsets, go on to Section 3.5.6 to initialize them.

3.5.5.2 Deleting RAIDsets

If you decide that you do not want the RAIDset that you just created, you can delete it and create a new one. If you decide to delete a RAIDset on which you have stored data, backup the data before deleting the RAIDset.

> **WARNING**
>
> Deleting a RAIDset destroys all data on the set.

To delete a RAIDset or sets, follow these steps:

1. At the RAIDset menu prompt, `Enter menu choice (1,7) [7] ?`
   You can type **2** to specify the RAIDset that you want to delete, or you can type **3** to delete all RAIDsets that you just created.

   If you type **3** and press the Return key, the utility displays the name of each RAIDset as it deletes it.

   If you Type: **2** and press the Return key, the utility prompts you to select one of the existing RAIDsets similar to the following, `Delete R1 (y/n/q) [n] ?`
2. Type **Y** and press the return key to select the RAIDset you want to delete. The utility displays the name of the RAIDset as it deletes it.
3. At the RAIDset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type **7** and press the Return key to return to the Main menu.

3.5.6 Initializing Storagesets

After you add a device to be used as an individual device, or create a storageset for the first time, you must initialize it.

> **WARNING**
>
> During the Initialization process, the CFMENU utility prompts you to specify a chunksize for stripesets and RAIDsets. Select the default chunksize (256) only. Do not use the CFMENU utility to set another chunksize, incorrect chunksize may result. To set a different chunksize, initialize the stripeset or RAIDset manually.

During initialization, the controller adds metadata on a small amount of the space available on the device and makes that space unavailable to the host. This metadata improves data reliability, error detection and recovery.

Manually initialize the last device or storageset in the subsystem to save a copy of the subsystem configuration on disk. See Section 3.6.7.3 to initialize manually.

To initialize a storageset, follow these steps:

1. At the Main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **5** and press the Return key.
   The Initialization menu appears similar to the screen shown in Figure 3–13.

Figure 3–13 The Initialization Menu

```
-------------------------------------CFMENU Configuration Menu Utility------------------------
                                 |Unconfig'd| Config'd   Device     Product     Stor.set Stor.set Chnk Trn- In-  Re-       W W
INITIALIZATION MENU:             | Dev.PTLs |   PTLs      Name        ID          Name    Typ/Sz  Size  sp. it'd duc  Unit P B
1. Initialize a device or        |          |disks:  130 DISK120 RZ29B (C) DEC                            N    N
   storageset (eligible entities |          |          ^ DISK130 RZ29B (C) DEC ^   S1     STR/3   256          Y
   marked with ^)                |          |STRPS:  100 DISK210 RZ29B (C) DEC            "       "            "
2. Return to main menu           |          |        110 DISK220 RZ29B (C) DEC            "       "            "
                                 |          |        200 DISK100 RZ29B (C) DEC ^   R1     RAD/3   256          Y    N
                                 |          |RAID5:  120 DISK110 RZ29B (C) DEC            "       "            "    "
WARNING!  Initialization of any  |          |        210 DISK200 RZ29B (C) DEC            "       "            "    "
device or storageset will        |          |        220
destroy all of its current data. |          |
                                 |          |
```

2. At the prompt, `Enter menu choice (1,2) [1] ?`
   Type: **`1`** and press the Return key.
   The utility prompts you to initialize the first uninitialized device or storageset that it locates.

3. Type: **`Y`** or **`N`** and press the Return key.

   Device or Mirrorset
   If you type **`N`**, the utility prompts you to initialize the next device or storageset that it locates.

   If you type **`Y`** to initialize a device or a mirrorset, the utility displays a message similar to that shown in Step 4.the following **`init disk100`**, initializes the device or storageset, and then prompts you to initialize the next device or storageset that it locates.

   Stripeset or RAIDset
   If you type **`N`**, the utility prompts you to initialize the next device or storageset that it locates.

   If you type **`Y`** to initialize a stripeset or RAIDset the utility displays the prompt, `Enter Chunksize (16..32768) {Default = Optimal based on characteristics of members] ?`

   > **WARNING**
   >
   > Accept the default chunksize. Do not try to change the chunksize for a stripeset or RAIDset using the CFMENU utility, incorrect chunksize may result. To set a different chunksize, initialize the stripeset or RAIDset manually.

   Press the Return key to accept the default chunksize and the utility displays a message similar to that shown in Step 4.

4. At the prompt: `Initialize this container with metadata space reserved on each device for SAVE_CONFIGURATION info (y,n) [n] ?`
   Type: **`Y`** to save a copy of the configuration information. To save the entire configuration, you need to select Y here when initializing one device or storageset only.
   The utility displays a message similar to the following, `init R1 CHUNKSIZE=DEFAULT SAVE_CONFIGURATION` , initializes the storageset, and then prompts you to initialize the next device or storageset that it locates.

5. Repeat Steps 3 and 4 until you initialize all of the devices or storagesets that you added to your system.
6. At the Initialization menu prompt, `Enter menu choice (1,2) [1] ?`
   Type: **2** and press the Return key.
   The Main menu appears.

If you created RAIDsets, mirrorsets, or striped mirrorsets, create a spareset of devices to be available to take the place of one of the devices in your RAIDset, mirrorset or striped mirrorset in the event that a device fails.

3.5.7 Adding a Spareset

Create sparesets to create a pool of devices available to the controller to use as a replacement device for one that fails in a RAIDset, mirrorset or striped mirrorset.

1. At the Main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **4** and press the Return key.
2. At the RAIDset menu prompt, `Enter menu choice (1,7) [7] ?`
   Type: **4** and press the Return key.
3. The Spareset/Failedset menu appears, similar to the screen shown in Figure 3–14.

Figure 3–14 The Spareset/Failedset Menu

```
----------------------------CFMENU Configuration Menu Utility----------------------------------
                             |Unconfig'd|  Config'd  Device     Product     Stor.set Stor.set Chnk Trn- In-  Re-        WP  WB
SPARESET/FAILEDSET MENU:     | Dev.PTLs |      PTLs   Name        ID          Name   Typ/Sz  Size sp. it'd duc Unit
1. Add a device to the       |----------|Strps: 100 DISK100 RZ29B (C) DEC *  S1      RAD/3   256       Y
   SPARESET (eligible devices|          |       110 DISK110 RZ29B (C) DEC            "       "        "
   marked by ^)              |          |       200 DISK200 RZ29B (C) DEC            "       "        "
2. Remove a device from the  |          |RAID5: 120 DISK120 RZ29B (C) DEC *  S1      STR/3   256       Y   N
   SPARESET                  |          |       210 DISK210 RZ29B (C) DEC            "       "        "   "
3. Remove a device from a    |          |       220 DISK220 RZ29B (C) DEC            "       "        "   "
   RAIDSET and place in the  |          |Spare: 130 DISK130 RZ29B (C) DEC
   FAILEDSET (eligible       |          |
   devices marked by *)      |          |
4. Remove a device from the  |          |
   FAILEDSET                 |          |
5. Return to RAIDSET menu    |          |
```

4. At the Spareset/Failedset menu prompt, `Enter menu choice (1,5) [5] ?`

   Type: **1** and press the Return key.
5. At the prompt, Add device DISK130 to SPARESET (y/n/q) [n] ?
   Type: **Y** or **N**.
   If you type Y to add the device to a spareset, the utility displays a message similar to the following: `add spareset DISK130`
6. The utility prompts you to add each available device to the spareset, then displays the Spareset/Failedset menu.
7. At the Spareset/Failedset menu prompt, `Enter menu choice (1,5) [5] ?`
   Type **5** and press the Return key to return to the Main menu.

3.5.8 Exiting CFMENU

Enter the last option from the main menu to stop CFMENU and return to the CLI. (You also may enter Ctrl/C or Ctrl/Y to abort CFMENU.)

3.6 Configuring Manually

To configure a device manually you need to do the following:

- Create storagesets
- Initialize storagesets
- Create spareset – if you create RAIDsets, mirrorsets, or striped mirrorsets, you want to create a spareset of devices to be available to take the place of one of the devices in your RAIDset, mirrorset or striped mirrorset in the event that a device fails.
- Add devices and storagesets to the host

3.6.1 Creating Storagesets

If you have not already done so, read Section 3.3 to map out what type of storagesets (Stripesets, Mirrorsets, RAIDsets or combinations) you want to use to configure your RAID Array 310. After mapping out your subsystem, follow Sections 3.6.3 to 3.6.7 to configure it manually. If you plan to create RAIDsets or Mirrorsets, don't forget to allocate some of your disks to the spareset, which will be available to replace a disk in the event that one fails from a RAIDset or Mirrorset.

NOTE

If you want to use disks that have already been used in a storageset, you must first delete the unit and then the storageset. and the associated disks. For example, if you created RAID1, Disk D100, you would use the following commands to delete the RAIDset and associated disk:

```
HSZ20> Delete D100

HSZ20> Delete RAID1
```

3.6.2 Verifying Storagesets

After you add or delete a storageset, use one of the following SHOW commands to verify that the CLI accepted the command that you entered.

- Type: **`show storageset`**
  The CLI displays a list of all the storagesets that the controller located.
- Type: `HSZ20>`**`show stripeset`**
  or **`show mirrorset`**
  or **`show raidset`**
  The CLI displays a list of all the stripesets, or mirrorsets or RAIDsets that the controller located.
- Type: `HSZ20>`**`show s1`**
  or **`show m1`**
  or **`show r1`**

  where: s1 = the name of a stripeset you created
  m1 is the name of a mirrorset you created
  r1 is the name of a RAIDset you created
  The CLI displays information about the stripeset, mirrorset, or RAIDset that you specified.

3.6.3 Adding/Deleting Stripesets

You can use CLI commands to add or delete stripesets.

3.6.3.1 Adding stripesets

Add stripesets by selecting the devices to use in each set. Stripesets must have between two and seven members.

Use the following CLI command to add a stripeset:

For example, type: `HSZ20>` **`Add stripeset stripe1 disk210 disk120 disk220`**

where: add = a CLI command
stripeset = the type of storageset
stripe1 = the stripeset name
disk210, disk120, disk220 = the devices you want to use for this stripeset

After you add the stripeset, see the following sections to create other storagesets, or go to Section 3.6.7 to initialize the Storagesets.

3.6.3.2 Deleting Stripesets

If you decide that you do not want a stripeset that you just created, you can delete it and create a new one. If you decide to delete a stripeset on which you have stored data, backup the data before deleting the Stripeset.

Use the following CLI commands to delete a stripeset.
Type: `HSZ20>` **`delete stripe1`**
where stripe1 = the name of a stripeset

3.6.4 Adding/Deleting Mirrorsets

You can use CLI commands to add and delete mirrorsets.

3.6.4.1 Adding Mirrorsets

Add Mirrorsets by selecting the devices to use in each set. Mirrorsets must have between two and six members. In addition, when you create a mirrorset, you can also change the settings for any or all of the parameters described in Table 3–8.

Table 3–8 Mirrorset Parameters

Parameter	Options	Default
Copy - Defines the speed at which the controller copies mirrorsets.	Normal The controller copies the mirrorset using minimal controller resources. Fast The controller copies the mirrorset using multiple resources, which speeds the copy and slows normal I/O processes to the host.	Normal
Policy - Defines the criteria which the controller uses to select the device to replace one that fails in a mirrorset.	Best_Performance Selects a device (from the spareset) to replace a device that fails in a mirrorset using the following criteria: 1. Achieves best performance of the mirrorset. 2. Most closely matches the sizes of the remaining devices in the mirrorset. Best_Fit Selects a device (from the spareset) to replace a device that fails in a mirrorset using the following criteria: 1. Most closely matches the sizes of the remaining devices in the mirrorset 2. Achieves best performance of the mirrorset. NoPolicy	Best_Performance
Read_Source - Defines the read algorithms that you want to use for a mirrorset.	Least_Busy The controller targets the member of a mirrorset with the least busy work queue for reads. Round_Robin The controller targets each member of a mirrorset for reads in sequential membership order. Preferred_Member The controller targets the member of a mirrorset that you specify for all reads.	Least_Busy

NOTE

You need to specify a mirrorset parameter only if you want to change its setting from the default.

Use the following CLI command syntax to add a mirrorset:

For example, type: `HSZ20>` **`Add MIRRORSET MIRR1 disk100 disk200 policy=best_fit`**

where: add = a CLI command
mirrorset = the storageset type
mirr1 = the storageset name
disk100, disk200 = the devices you want to use for this mirrorset
policy = the policy parameter being set to best_fit rather than the default

After you add the mirrorset, see the following sections to create other storagesets, or go to Section 3.6.7 to initialize the mirrorsets.

3.6.4.2 Deleting Mirrorsets

If you decide that you do not want a mirrorset that you just created, you can delete it and create a new one. If you decide to delete a mirrorset on which you have stored data, backup the data before deleting the mirrorset.

Use the following CLI commands to delete a mirrorset:

For example, type: `HSZ20>` **`delete mirror1`**
where mirror1 is the name of a mirrorset

3.6.5 Adding Striped Mirrorsets

Add Stripe-Mirrorsets by selecting the Mirrorsets to use in a stripeset.

Use the following CLI command syntax to add a striped mirrorset:

For example, type: `HSZ20>` **`add stripeset stripe2 mirr1 mirr2 mirr3`**

where: add = a CLI command
stripeset = the storageset type
stripe2 = the storageset name
mirr1, mirr2, mirr3 = the mirrorsets you want to stripe
After you add the stripe-mirrorset, go to Section 3.6.7. to initialize it.

3.6.6 Adding/Deleting RAIDsets

You can use CLI commands to add and delete mirrorsets.

3.6.6.1 Adding RAIDsets

Add RAIDsets by selecting the devices to use in each set. RAIDsets must have between three and seven members. In addition, when you create a RAIDset, you can also change the settings for any or all of the parameters described in Table 3–9.

Table 3–9 RAIDset Parameters

Parameter	Options	Default
Policy - Defines the criteria which the controller uses to select the device to replace one that fails in a RAIDset.	Best_Performance Selects a device (from the spareset) to replace a device that fails in a RAIDset using the following criteria: 1. Achieves best performance of the RAIDset. 2. Most closely matches the sizes of the remaining devices in the RAIDset. Best_Fit Selects a device (from the spareset) to replace a device that fails in a RAIDset using the following criteria: 1. Most closely matches the sizes of the remaining devices in the RAIDset 2. Achieves best performance of the RAIDset. NoPolicy Does not select a replacement; allows the RAIDset to operate in a reduced state.	Best_Performance
Reconstruct – Specifies the rate (speed) at which the controller reconstructs a RAIDset after adding a replacement drive	Normal Balances overall performance of controller against RAIDset reconstruction operations Fast Gives priority of controller operation to reconstruction of RAIDset and may reduce performance of other controller operations. NoReconstruct	Normal
Reduced – Specifies whether a RAIDset is missing a member.	NoReduced Specifies that RAIDset being added contains the minimum number of members. Reduced Specifies that a RAIDset being add is already missing a member	NoReduced

NOTE

You need to specify a RAIDset parameter only if you want to change its setting from the default.

Use the following CLI command syntax to add a RAIDset:

For example, type: `HSZ20>` **`Add RAIDset raid1 disk100 disk110 disk200 policy=best_performance`**

where: add = a CLI command
RAIDset = the storageset type
raid1 = the storageset name
disk100, disk110, disk200 = the devices you want to use for this RAIDset
policy = the policy parameter being set to best_fit rather than the default

After you add the RAIDset, see the following sections to create other storagesets, or go to Section 3.6.7 to initialize the RAIDset.

3.6.6.2 Deleting RAIDsets

If you decide that you do not want a RAIDset that you just created, you can delete it and create a new one. If you decide to delete a RAIDset on which you have stored data, backup the data before deleting the RAIDset.

Use the following CLI commands to delete a RAIDset:

For example, type: `HSZ20>` **`delete RAID1`**
where RAID1 is the name of a RAIDset

3.6.7 Initializing Devices and Storagesets

> **CAUTION**
>
> The INITIALIZE command destroys all data on a device. (Do not initialize a tape device.)

After you add a device to be used as an individual device, or create a storageset for the first time, you must initialize it.

Save a copy of the subsystem configuration on disk by initializing the last device or storageset in the subsystem using the parameters specified in Section 3.6.7.3.

During initialization, the controller adds metadata on a small amount of the space available on the device and makes that space unavailable to the host. This metadata improves data reliability, error detection and recovery. The initialization process varies slightly for a device or a storageset.

3.6.7.1 Initializing a Device or Mirrorset

Use the following CLI command syntax to initialize a device or storageset. For example, type: `HSZ20>` **`Initialize disk100`**

where: Initialize = a CLI command
disk100 = the name of the device or mirrorset you want to initialize.

3.6.7.2 Initializing a Stripeset, RAIDset or Striped Mirrorset

When you initialize a stripeset, RAIDset or striped mirrorset you can specify a new chunksize to change the chunksize from the default of 256 blocks. A chunksize is the number of blocks of data that the controller transfers at one time.

Valid chunksizes range as follows:

- For RAIDsets: 16 blocks minimum to a maximum number of blocks, depending upon number of members of RAIDset.
  3 members = 1024 blocks maximum
  4 members = 682 blocks maximum
  5 members = 512 blocks maximum
  6 members = 409 blocks maximum
  7 members = 341 blocks maximum
- For Stripesets: 16 blocks minimum. Stripesets have no effective chunksize maximum as the maximum chunksize is equivalent to 2 gigabytes or the potential size of a whole device.

Use a larger chunksize for applications that make many I/O requests. Use a smaller chunksize for applications that make relatively few I/O requests but need to move large amounts of data with each request.

Use the following CLI command to initialize a Stripeset or RAIDset, for example, type:
HSZ20> **`initialize raid1 chunksize=1024`**

where: Initialize = a CLI command
RAID1 = the name of the stripeset or RAIDset you want to initialize.
chunksize = sets the chunksize for this RAIDset to 128

> **NOTE**
>
> It may take up to three minutes for a RAIDset or stripeset to initialize.

3.6.7.3 Saving Copies of the Configuration

Use the following initialize command to save a copy of the controller configuration on a device or storageset in the subsystem. The controller automatically updates the saved copy of the configuration whenever the configuration changes.

We recommend keeping a copy of the configuration on at least two devices or storagesets.

To save a copy of the RAID Array 310 configuration on disk, initialize a device or storageset that you add to your subsystem as follows:

Type: **`initialize disk100 save_configuration`**

where: Initialize = a CLI command

DISK100 = the name of the device or storageset you want to initialize
Save_configuration = the parameter that saves the configuration of this RAID Array 310 subsystem on the device or storageset that you specified

The controller places a copy of the configuration onto the specified device or storageset and automatically updates this saved copy whenever the configuration changes. To ensure availability of a copy of the configuration, save the configuration on at least two devices. See Section 4.16, Replacing a Failed Controller in Chapter 4 to restore the configuration from disk.

3.6.8 Adding/Deleting Devices to the Spareset

Add devices to the spareset to create a pool of devices available to the controller to use as a replacement device for one that fails in a RAIDset, mirrorset or striped mirrorset. You can use CLI commands to add and delete sparesets.

3.6.8.1 Adding Devices to the Spareset

Use the following CLI command to add a device to the spareset:
Type: **`add spareset disk100`**

where: add = a CLI command
spareset = the type of storageset
disk100 = the device you want to use as a spare

You can only add one device at a time. If you want to use more than one device, you need to repeat the add spareset command for each device you want to use as a spare.

3.6.8.2 Deleting a Device from the Spareset

Use the following CLI command to delete a device from the spareset:
Type: **`delete spareset disk100 disk210`**
where: add = a CLI command
spareset = the type of storageset
disk100, disk210 = the devices you want to remove from a spareset. You can specify one device or many to remove from a spareset

3.7 Adding Devices and Storagesets to the Host

As you added/named each device to let the controller know it is there, you must now add/name each device or storageset to let the host know it is there. You make a device or storageset known to the host by adding it as a unique unit.

You can add units either automatically, using the CFMENU utility or manually, using CLI commands.

> **NOTE**
>
> If the write-back cache battery condition is low, you can add a RAIDset as a unit, but you cannot access that RAIDset unless you had set the controller cache policy to B. For more information on the cache policy see Table 3–1.

3.7.1 Adding Automatically

To add a unit automatically, follow these steps:

1. At the Main menu prompt, `Enter menu choice (1,8) [8] ?`
   Type: **`6`** and press the Return key.
   The Unit menu appears similar to the screen shown in Figure 3–15.

Figure 3–15 The Unit Menu

```
-----------------------------------CFMENU Configuration Menu Utility----------------------------------
                             | Unconfig'd | Config'd   Device    Product      Stor.set Stor.set Chnk Trn- In-  Re-       W W
UNIT MENU:                   | Dev.PTLs   |  PTLs      Name       ID           Name    Typ/Sz   Size sp.  it'd duc Unit  P B
1. Create a unit (eligible   |            |Strps:  100  DISK100 RZ29B (C) DEC *  S1      RAD/3    256       Y         D100
   entities marked by ^)     |            |        110  DISK110 RZ29B (C) DEC               "       "        "
2. Delete a unit (eligible   |            |        200  DISK200 RZ29B (C) DEC               "       "        "
   units marked by *)        |            |RAID5:  120  DISK120 RZ29B (C) DEC *  S1      STR/3    256       Y    N    D200  N N
3. Return to main menu       |            |        210  DISK210 RZ29B (C) DEC               "       "        "    "
                             |            |        220  DISK220 RZ29B (C) DEC               "       "        "    "
                             |            |Spare:  130  DISK130 RZ29B (C) DEC
```

2. At the prompt, `Enter menu choice (1,3) [1] ?`
   Type: **`1`** and press the Return key.
3. The utility prompts you to add the first device or storageset that it locates as a unit.
   Type: **`Y`** to add that storageset, **`N`** to add another storageset, or **`Q`** to exit this process and press the Return key.
4. At the prompt, `Enter unit number for new unit:`
   Type **`a number for the unit`** and press the Return key.
   To assign a unit number , you must first determine what target SCSI IDs have been set for the controller. You can then assign a unit number from 0 –7 for each of the target

SCSI IDs set for the controller. For best performance, assign as many units as possible to unique target SCSI IDs. (You can have up to four target SCSI IDS.)

For example, if you set controller target SCSI IDs 0 – 3, you can then create six units named as follows:

Target SCSI ID	Unit Number Range	Unit Names
0	000 – 007	D000, D001
1	100 – 107	D100, 102
2	200 – 201	D200
3	300 – 301	D30

Although this example shows two units assigned to target SCSI IDs 0 and 1, for best performance do not assign the second unit to a particular target SCSI ID until after you assign one unit to each available target SCSI ID.

5. The utility now prompts you to specify a series of parameters for the unit. Table 3–10 describes these parameters.
6. At the prompt, `Set WRITE_PROTECT on unit (y/n) [n] ?`
   Press the Return key to accept the default of no write protect.
7. At the prompt, `Set RUN on unit (answering NO disables drives' ability to be spun up) (y/n) [y] ?`
   Press the Return key to accept the default of set RUN on unit.
8. At the prompt, `Enable WRITEBACK cache for this unit (y/n) [n] ?`
   Press the Return key to accept the default of set to WRITETHROUGH CACHE.
   The utility now adds a RAIDset as a unit. For other storagesets the utility requires one more step prior to adding the storageset as a unit.
9. At the prompt, `Enable READ_CACHE for this unit (y/n) [y] ?`
   Type **`Y`** to enable READ_CACHE, or type **`N`** to disable READ_CACHE. The utility displays a message similar to the following, `add unit 100 M1 NOWRITEBACK_CACHE READ_CACHE`, and adds the storageset as the unit number which you specified.
10. At the Unit menu prompt, `Enter menu choice (1,3) [1] ?`
    Type: **`3`** and press the Return key.
    The Main menu appears. The CFMENU Main Menu screen displays the unit number for this storageset.

WARNING

Although the utility prompts you to specify a WRITEBACK_CACHE option, only use the default setting when using the CFMENU utility. To set another WRITEBACK_OPTION see Section 3.7.2.

Table 3–10 Unit Parameters

Parameter	Options	Default
WRITE_PROTECT	WRITE_PROTECT	NOWRITE_PROTECT

Prevents the host from writing data to the unit. You can set a unit to write_protect at any time after you create the unit and store data on it.	NOWRITE_PROTECT	
RUN Allows the host to access the unit. You can set the unit to NORUN at any time after you create the unit to prevent the host from accessing it.	RUN NORUN	RUN

Table 3–10 Unit Parameters (cont.)

Parameter	**Options**	**Default**
WRITEBACK_ CACHE Write cache determines the method by which the host writes data to the units in the subsystem. The two write cache options are writeback and write through. Use WRITEBACK_CACHE to increase performance of host write requests. When the host requests a write operation, the cache writes the host's data first to the cache memory completing the host's request quickly. It performs the slower operation of flushing the data to the external storage device at a later time. The host sees the write operation as complete when the data have reached the cache. Use WRITETHROUGH_CACHE for the host to write data directly to the unit. The host sees the write operation as complete only after the external storage device has been updated.	WRITETHROUGH_ CACHE WRITEBACK_ CACHE	WRITETHROUGH_ CACHE
READ_CACHE Read cache determines the method by which the host reads data from the units in the subsystem. The two read cache options are READ_CACHE and NOREAD_CACHE. Use READ_CACHE for the host to read data from a high speed memory buffer between the unit and the host. A read cache increases the controller's effective device access speed by satisfying host read requests from its local cache memory when possible, instead of from external storage devices. Use NOREAD_CACHE for the host to read data directly from the unit.	READ_CACHE NOREAD_CACHE	READ_CACHE
MAXIMUM_CACHE_TRANSFER Specifies in blocks the maximum transfer size that the controller caches. The controller does not cache any transfer larger than this size.	16	1-1024 Blocks

NOTE

Readback not available for RAIDset

This completes the steps required to configure your subsystem. In addition, see the Getting Started guide for the operating system running on your host system to begin using the units that you created.

3.7.2 Adding Manually

The process of adding units varies slightly for devices and storagesets.

3.7.2.1 Adding Devices as Units

Use the following CLI command to add a device as a unit, for example,
type: `HSZ20>` **`add unit d100 disk100`**

where: add = a CLI command
unit = what you are adding
d100 = a unique identifier for this unit.
To assign a unit number , you must first determine what target SCSI IDs have been set for the controller. You can then assign a unit number from 0 –7 for each of the target SCSI IDs set for the controller. For best performance, assign as many units as possible to unique target SCSI IDs. (You can have up to four target SCSI IDS.)

For example, if you set controller target SCSI IDs 0 – 3, you can then create six units named as follows:

Target SCSI ID	**Unit Number Range**	**Unit Names**
0	000 – 007	D000, D001
1	100 – 107	D100, 102
2	200 – 201	D200
3	300 – 301	D30

Although this example shows two units assigned to target SCSI IDs 0 and 1, for best performance do not assign the second unit to a particular target SCSI ID until after you assign one unit to each available target SCSI ID.

disk100 = the device you want to assign to a unit

3.7.2.2 Adding Storagesets as Units

When you add a storageset as a unit, you can specify settings for the parameters that control its operation as described in Table 3–10. If you do not specify new settings, the CLI creates the unit using the default parameter settings You can change most of the settings for a unit at anytime.

WARNING

You cannot set the following parameters for the following types of devices or storagesets:

READ_CACHE	RAIDset
WRITEBACK_CACHE	Transportable Device

Use the following CLI command to add a storageset as a unit:
Type: `HSZ20>` **`add unit d0 raid1 writeback_cache`**

where: add = a CLI command
unit = what you are adding
d0 = a unique identifier for this unit. Valid unit numbers include 100 – 107 and 200 – 207
raid1 = the storageset you want to assign to a unit
writeback_cache = a parameter setting to which you want to set this unit

This completes the steps required to configure your subsystem. If you created RAIDsets, mirrorsets, or striped mirrorsets, create a spareset of devices to be available to take the place of one of the devices in your RAIDset, mirrorset or striped mirrorset in the event that a device fails. In addition, see the Getting Started guide for the operating system running on your host system to begin using the units that you created.

3.8 Verifying your RAID Configuration

To verify and record your configuration for future reference, verify the units, storagesets and devices you have configured.

3.8.1 Verifying Units

Use the following CLI command to verify units.

Type: `HSZ20>` **`show units`**

The controller displays a list similar to the following:

```
LUN      USES
----------------
D100     M1
D101     M2
D102     M3
```

3.8.2 Verifying Storagesets

Use the following CLI command to verify storagesets.

Type: `HSZ20>` **`show storagesets`**

The controller displays a list similar to the following:

```
Name             storageset       Uses             Used by
---------------------------------------------------------
RAID1            raidset          DISK100          D100
                                  DISK110
                                  DISK200
RAID             raidset          DISK120          D101
                                  DISK130
                                  DISK210
SPARESET         spareset         DISK220
FAILEDSET        failedset
```

This report does not include individual devices.

```
Name             StorageSet       Uses             Used by
----------------------------------------------------------
RAID1            raidset          DISK100          D100
                                  DISK110
                                  DISK200
RAID             raidset          DISK120          D101
                                  DISK130
                                  DISK210
SPARESET         spareset         DISK220
FAILEDSET        failedset
```

3.8.3 Verifying Devices

Use the following CLI command to verify units.

Type: `HSZ20>` **`show devices`**

The controller displays a list similar to the following:

```
Name      Type   Port  Targ  Lun    Used by
---------------------------------------------------------------
DISK100   disk   1     0     0      M1
DISK110   disk   1     1     0      M2
DISK120   disk   1     2     0      M3
DISK130   disk   1     3     0
DISK200   disk   2     0     0      M1
DISK210   disk   2     1     0      M2
DISK220   disk   2     2     0      M3
```

Record the configuration information on a piece of paper. In the event of a controller failure, the information that you recorded will assist you in reconstructing the configuration of your RAID Array 310.

See the appropriate getting started guide to prepare your subsystem for access by the operating system in use by your host computer.

3.9 Automatically Replenishing the Spareset with Devices Replaced in the Failedset

To aid in replacing failed disk drives, the RAID Array 310 allows you to enable the failedset so that when you replace a device in the failedset, the RAID Array 310 automatically places the new device in the SPARESET.

Once you enable this option, when you remove a device from the failedset and install a new device (that does not contain any metadata) in the same slot, the controller will automatically move that device in the configuration from the FAILEDSET to the SPARESET. It is then available for any RAIDset or mirrorset that is missing a member.

The feature is disabled by default. To enable this feature, type:

```
HSZ20> SET FAILEDSET AUTOSPARE
```

To disable the feature:

```
HSZ20> SET FAILEDSET NOAUTOSPARE
```

You can view the current setting with the SHOW FAILEDSET command:

```
HSZ20> SHOW FAILEDSET

Name         Storageset         Uses          Used by
------------------------------------------------------
FAILEDSET    failedset
        Switches:
          AUTOSPARE
```

The Auto New Spare feature only operates when the newly-inserted device does not contain any metadata, such as a device from the factory. You can use the TRANSPORTABLE function to initialize a used device so that it no longer contains metadata by issuing the following CLI commands:

```
HSZ20> SET disk-name TRANSPORTABLE

HSZ20> INITIALIZE disk-name
```

To initialize additional devices, simply remove the first device and replace it with another, then retype the INITIALIZE command. You do not need to reenter the ADD DISK or SET TRANSPORTABLE commands as long as you use the same SCSI-location.

When the controller moves the device into the spareset, it writes some metadata on it and the device is no longer transportable. The controller writes additional metadata on a device when the device moves from the spareset into a RAIDset or mirrorset.

3.10 CFMENU Messages

This section lists the messages, other than the standard CLI messages, that CFMENU displays. However, you will mostly see messages sent by the CLI, as described in Appendix A.

Message:

CFMENU cannot complete request without exceeding array boundary.

Explanation: CFMENU detected an unexpected condition which would exceed an array boundary and possibly require controller reinitialization, so it aborted your request.

Message:

...CFMENU is updating its configuration tables. Please be patient...

Explanation: CFMENU is polling the SCSI ports to see what physical devices are in place, as well as checking the configuration information.

Message:

MIRROR license is not enabled; cannot create mirrorset.

Explanation: You may not configure any mirrorsets unless the Disk Mirroring license has been enabled by running the FLS utility.

Message:

Not enough eligible devices to complete the storageset.

Explanation: You chose to create a stripeset or a RAIDset and specified how many members to use, but there are not enough eligible devices to make up a storageset of this size. Eligible devices are disks that have the NOTRANSPORTABLE switch set, and that are not already used in any higher-level configuration such as a unit, storageset, spareset, or failedset.

Message:

Not enough members specified for a non-reduced RAIDset

Explanation: You chose to create a RAIDset and specified that it is not a previously reduced RAIDset. However, when choosing how many members to add to the RAIDset, you specified a number that is only legal for a reduced RAIDset and is too low for a nonreduced RAIDset.

Message:

...Polling for unconfigured devices...

Explanation: CFMENU is polling the SCSI ports to see what physical devices are in place.

Message:

Port *port#* is blocked. No devices will be configured on port *port#*

Explanation: In order to check each device bus to discover what devices are present, firmware must also see if any port is currently blocked. A port can be blocked for various reasons, such as when its bus is quiesced. When a port is blocked, CFMENU will not access devices on that bus.

Message:

```
RAID5 license is not enabled; cannot create RAID5 set.
```

Explanation: You may not configure any RAIDsets unless the RAID5 license has been enabled by running the FLS utility.

Message:

```
Received user request to terminate CFMENU...
```

Explanation: You pressed Ctrl/C or Ctrl/Y to abort CFMENU.

Message:

```
There are no devices available to use as a replacement.
```

Explanation: You chose to replace a member of a reduced RAIDset, but there are no disks eligible to use for the replacement member. Eligible disks must have the NOTRANSPORTABLE switch set and may not be part of any higher-level configuration such as units, storagesets, or the spareset or failedset.

Message:

```
There are no devices eligible to be added to the
configuration.
```

Explanation: You chose to add a device, but there are no devices available to add. The only devices that are eligible to be added are devices that CFMENU has detected as being physically present on a SCSI port and that are not already configured as devices on the controller.

Message:

```
There are no devices eligible to be added to the spareset.
```

Explanation: You chose to add a device to the spareset, but no devices are eligible. The only devices that may be added to the spareset are disks that have the NOTRANSPORTABLE switch set, and that are not already used in any higher-level configuration such as a unit, storageset, spareset, or failedset.

Message:

```
There are no devices eligible to be mirrored.
```

Explanation: You chose to mirror a device but there are no devices eligible to be mirrored. Eligible devices are disks that have the NOTRANSPORTABLE switch set and are configured as units or as part of a stripeset unit.

Message:

```
There are no devices eligible to be unmirrored.
```

Explanation: You chose to unmirror a device but there are no devices eligible to be unmirrored. Eligible devices are the only members of 1-member mirrorsets that are configured as units or as part of a stripeset unit.

Message:

`There are no devices in the failedset.`

Explanation: You chose to delete devices from the failedset but the failedset currently is empty.

Message:

`There are no devices in the spareset.`

Explanation: You chose to delete devices from the spareset but the spareset currently is empty.

Message:

`There are no entities eligible for initialization.`

Explanation: You chose to initialize a device or storageset, but there are currently no devices or storagesets that are eligible to be initialized. Tapes, CD–ROMs, loaders, and passthrough devices may not be initialized. Optical memory devices, stripesets, and RAIDsets may be initialized, but only if they are not already configured as a unit. Disks may be initialized only if they are not already configured as a unit or as part of a storageset, spareset, or failedset.

Message:

`There are no entities eligible to be added as units.`

Explanation: You chose to add a unit, but there are no devices or storagesets that are eligible to become units. Disks, optical memory, stripesets, and RAIDsets must first be initialized before they can be added as units. Disks may not be made into units if they are currently in the spareset or the failedset.

Message:

`There are no mirrorsets eligible for changing membership count.`

Explanation: You chose to change the membership count of a mirrorset, but there are no mirrorsets eligible for this operation. To be eligible, a mirrorset must be associated with a unit and must not already have the maximum number of members allowed for a mirrorset.

Message:

`There are no reduced` *`RAIDset/mirrorset`* `units with NOPOLICY set.`

Explanation: You chose to replace a member of a reduced RAIDset. CFMENU is unable to find any RAIDsets that are eligible for a manual replacement. In order to be eligible, the RAIDset must be configured as a unit, it must be in a reduced state, and it must have the NOPOLICY switch set.

Message:

```
There are no storageset units with members that can be moved
to the FAILEDSET.
```

Explanation: You chose to move a device from a storageset to the FAILEDSET, but there are no devices eligible for that operation. Eligible devices must be members of RAIDsets or mirrorsets that are associated with a unit. If the device is a member of a RAIDset, it must not be a reduced RAIDSET. If the device is a member of a mirrorset, removing the device from the mirrorset must leave at least one member of the mirrorset in a normal state.

Message:

```
There are no units to delete.
```

Explanation: You chose to delete a unit, but there are no units configured on the controller.

Message:

```
Unable to allocate memory, CFMENU terminating.
```

Explanation: There is not enough memory available for CFMENU to run.

Message:

```
User has not picked enough eligible devices to complete the
storageset.
```

Explanation: You chose to create a stripeset or a RAIDset and specified how many members to use, but when CFMENU prompted for devices, you did not select enough to complete the storageset.

Message:

```
Waiting for completion of CLI command...
```

Explanation: Some CLI commands take a long time to complete, such as initializing a large RAIDset or adding a tape drive that needs to have the tape rewound. CFMENU prints out this message periodically to inform you that it is still waiting for the last CLI command to finish.

4

Maintaining the RAID Array 310 Subsystem

This chapter describes procedures required to maintain the RAID Array 310 Subsystem through the onboard controller firmware.

4.1 Introduction

Maintaining your subsystem requires a variety of physical activities and the use of the onboard controller firmware.

This chapter describes activities to maintain your subsystem using the onboard controller firmware. These activities include:

- Viewing Subsystem Configuration
- Viewing Device and Storageset Status
- Matching Device Displayed on CLI with Physical Device in the Subsystem Cabinet
- Recovering after a Device Fails from a Storageset
- Replacing a Failed Device with a Device in Good Working Order
- Changing Storageset Parameter Settings
- Removing a Member of a RAIDset
- Removing a Member Device from a Mirrorset
- Unmirroring a One Member Mirrorset
- Creating Two Stripesets out of a Striped Mirrorset
- Obtaining Snapshot Copies of Data From Mirrorsets
- Upgrading Firmware
- Installing Firmware Code Patches
- Setting Feature Licenses
- Replacing a Failed Controller

See the Hardware manual for information on maintaining the physical aspects of the subsystem.

NOTE

The controller maintains maps of devices and locations and uses those maps for communicating with devices. If you add, move or change devices with the controller power turned-off, and without first changing the controller configuration online, the controller cannot work with the changed devices when it returns to service.

4.2 Viewing Subsystem Configuration

Use the Command Line Interface (CLI) SHOW and SHOW X FULL commands to view the subsystem configuration.

You can use the SHOW and SHOW X FULL (where X = any of the following subsystem elements that you can view) to view the following:

- Devices
- All Storagesets
- Stripesets
- RAIDsets
- Mirrorsets
- Striped Mirrorsets
- The Spareset
- The Failedset
- Controller Parameters

The SHOW command specified with any type of storageset displays information in the following format:

```
Name          Storageset    Uses          Used by
-------------------------------------------------------------
RAID1         raidset       DISK100       D100
                            DISK110
                            DISK210
```

Table 4–1 Describes the information displayed by the show command format

Table 4–1 Show Command Format Display

Show Command Heading	Description
Name	Lists names of existing storagesets.
Storageset	Identifies type of storageset.
Uses	Identifies devices that makeup the storageset.
Used by	Identifies the unit number assigned to this storageset. Unit numbers allow the host to recognize the storageset.
Port	The channel on the RAID 310 controller to which this device is connected.
Tar (Target)	The device's SCSI address, determined by the cabinet slot in which you installed the device.
LUN	The LUN number for each device installed in a cabinet slot is 0.

The SHOW command specified with any of the subsystem elements listed above and the FULL parameter displays the following additional information:

- Switch or parameter settings
- State of each storageset
- Size in blocks of each storageset

Table 4–2 describes the information that the utility displays for the various configurations.

Table 4–2 Device and Storageset Information Displayed by the Show Command.

Displayed Information	Description	Applies To
Switches		
Chunksize	A chunksize is the number of blocks of data that the controller transfers at one time. Valid chunksizes include 16 to 682 blocks. Applications that make many I/O requests use a larger chunksize. Applications that make relatively few I/O requests but need to move large amounts of data with each request use smaller chunksizes.	Storagesets
Policy	Defines the criteria which the controller uses to select the device to replace one that fails in a mirrorset. See Table 3–8 for a description of the policy options.	Mirrorsets RAIDsets
Copy	Defines the speed at which the controller copies mirrorsets. See Table 3–8 for a description of the copy options.	Mirrorset

Table 4–2 Device and Storageset Information Displayed by the Show command (cont.)

Displayed Information	Description	Applies To
Switches		
READ_SOURCE	Defines the read algorithms that you want to use for a mirrorset. See Table 3–8 for a description of the read_source options.	Mirrorset
Membership	Identifies the number of members in this mirrorset.	Mirrorset
Reconstruct	Specifies the rate (speed) at which the controller reconstructs a RAIDset after adding a replacement drive. See Table 3–7 or 4–6 for a description of reconstruct options.	RAIDset
Reduced	Specifies whether a RAIDset is missing a member. See Table 3–7, or 4–5 for a description of reduced options.	RAIDset
Transportable	The transportable/notransportable setting specifies whether you can move devices from your RAID Array 310 to another subsystem. Digital does not recommend moving devices from your RAID Array 310 and thus setting your devices notransportable. See Section 3.2.1 for more information about transportability.	Devices
State		
	States indicate the operating condition of the storageset based on the operating conditions of each of the member devices of the storageset. See Section 4.3 for a description of the various states.	Storagesets
Size		
	The size of the storageset in blocks.	Storagesets
Miscellaneous		
DEC RZ29B (C) DEC 0012	Device manufacture information.	Devices

4.2.1 Viewing All Devices

You can view either a list of devices or a list of devices including their switches, states and sizes.

4.2.1.1 Viewing a List of Devices

To view a simple list of devices, type: `HSZ20>` **`show devices`**

The CLI displays information similar to the following:

```
Name           Type           Port Targ  Lun           Used by
------------------------------------------------------------
DISK100        disk              1    0    0           STRIPE1
DISK110        disk              1    1    0           STRIPE1
DISK120        disk              1    2    0           MIRROR1
DISK130        disk              1    3    0           MIRROR1
DISK200        disk              2    0    0           STRIPE1
DISK210        disk              2    1    0           STRIPE1
DISK220        disk              2    2    0           SPARESET
```

4.2.1.2 Viewing All Device Information

To view all information about devices, type: `HSZ20>` **`show devices full`**

The CLI displays information similar to the following:

```
Name           Type           Port Targ  Lun           Used by
------------------------------------------------------------
DISK100        disk              1    0    0
          DEC      RZ29B    (C)  DEC 0012
DISK110        disk              1    1    0
          DEC      RZ29B    (C)  DEC 0012
DISK120        disk              1    2    0
          DEC      RZ29B    (C)  DEC 0012
```

4.2.2 Viewing All Storagesets

You can view either a list of storagesets or a list of storagesets including their switches, states and sizes.

4.2.2.1 Viewing a list of Storagesets

To view a simple list of storagesets, type: `HSZ20>` **`show storagesets`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
-------------------------------------------------------------
RAID1         raidset       DISK100       D100
                            DISK110
                            DISK210
SPARESET      spareset
FAILEDSET     failedset
```

4.2.2.2 Viewing All Storageset Information

To view all information about all storagesets,
type: `HSZ20>` **`show storagesets full`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
---------------------------------------------------------
STRIPE0       stripeset     DISK100       D100
                            DISK110
                            DISK200
     Switches:
       CHUNKSIZE = 128 blocks
     State:
       NORMAL
       DISK100 (member 0) is NORMAL
       DISK110 (member 1) is NORMAL
       DISK200 (member 2) is NORMAL
     Size: 2050353
RAID0         raidset       DISK120       D101
                            DISK210
                            DISK220
     Switches:
       POLICY (for replacement) = BEST_PERFORMANCE
       RECONSTRUCT (priority)  = NORMAL
       CHUNKSIZE = 128 blocks
     State:
       NORMAL
       DISK100
       DISK210
       DISK220
SPARESET      spareset      DISK130
FAILEDSET     failedset
```

4.2.3 Viewing Stripesets

You can view stripesets in the following:

- A list of all stripesets
- A list of all stripesets including their switches, states and sizes
- A specific stripeset

4.2.3.1 Viewing a list of Stripesets

To view a list of stripesets, type: `HSZ20>` **`show stripesets`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
----------------------------------------------------------
STRIPE0       stripeset     DISK100       D100
                            DISK110
                            DISK200
STRIPE1       stripeset     DISK120       D101
                            DISK210
                            DISK220
```

4.2.3.2 Viewing All Stripeset Information

To view all information about all stripesets, type: `HSZ20>` **`show stripesets full`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
----------------------------------------------------------
STRIPE0       stripeset     DISK100       D100
                            DISK110
                            DISK200
     Switches:
        CHUNKSIZE = 128 blocks
     State:
      NORMAL
      DISK100   (member 0) is NORMAL
      DISK110   (member 1) is NORMAL
      DISK200   (member 2) is NORMAL
     Size: 2050353
STRIPE1       stripeset     DISK120       D200
                            DISK210
                            DISK220
     Switches:
        CHUNKSIZE = 128 blocks
     State:
      NORMAL
      DISK120   (member 0) is NORMAL
      DISK210   (member 1) is NORMAL
      DISK220   (member 2) is NORMAL
     Size: 2050353
```

4.2.3.3 Viewing a Stripeset

To view a specific stripeset, type: `HSZ20>` **`show stripe0`**
where stripe0= the name of a stripeset

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
---------------------------------------------------------
STRIPE0       stripeset     DISK100       D100
                            DISK110
                            DISK200
     Switches:
       CHUNKSIZE = 256 blocks
     State:
      NORMAL
      DISK100 is NORMAL
      DISK110 is NORMAL
      DISK200 is NORMAL
     Size: 12328410
```

4.2.4 Viewing Mirrorsets

You can view mirrorsets in the following ways:

- A list of all mirrorsets
- A list of all mirrorsets including their switches, states and sizes
- A specific mirrorset including its switches, states and sizes

4.2.4.1 Viewing a list of Mirrorsets

To view a list mirrorsets, type: `HSZ20>` **`show mirrorsets`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
---------------------------------------------------------
MIRROR1       mirrorset     DISK100       D100
                            DISK200
MIRROR2       mirrorset     DISK110       D200
                            DISK210
```

4.2.4.2 Viewing All Mirrorset Information

To view all information about all mirrorsets, type: `HSZ20>` **`show mirrorsets full`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
---------------------------------------------------------
MIRROR1       mirrorset     DISK100       D100
                            DISK200
     Switches:
       POLICY (for replacement) = BEST_PERFORMANCE
       COPY (Priority) = NORMAL
       READ_SOURCE = LEAST_BUSY
       MEMBERSHIP = 2, 1 MEMBER PRESENT
     State:
      DISK100   (member 0) is NORMAL
     Size: 2050353
MIRROR2       mirrorset     DISK110       D102
                            DISK210
     Switches:
       POLICY (for replacement) = BEST_PERFORMANCE
       COPY (Priority) = NORMAL
       READ_SOURCE = LEAST_BUSY
       MEMBERSHIP = 2, 1 MEMBER PRESENT
     State:
      DISK110   (member 0) is NORMAL
     Size: 2050353
```

4.2.4.3 Viewing a Mirrorset

To view a specific mirrorset, type: `HSZ20>` **`show mirror1`**
where mirror1 = the name of a stripeset

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
---------------------------------------------------------
MIRROR1       mirrorset     DISK100       D100
                            DISK200
     Switches:
       POLICY (for replacement) = BEST_PERFORMANCE
       COPY (Priority) = NORMAL
       READ_SOURCE = LEAST_BUSY
       MEMBERSHIP = 2, 1 MEMBER PRESENT
     State:
      DISK100   (member 0) is NORMAL
     Size: 2050353
```

4.2.5 Viewing RAIDsets

You can view RAIDsets in the following ways:

- A list of all RAIDsets
- A list of all RAIDsets including their switches, states and sizes
- A specific RAIDset

4.2.5.1 Viewing a list of RAIDsets

To view a list RAIDsets, type: `HSZ20>` **`show raidsets`**

The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
----------------------------------------------------------
RAID1         raidset       DISK100       D100
                            DISK110
                            DISK200
RAID2         raidset       DISK120       D200
                            DISK210
                            DISK220
```

4.2.5.2 Viewing All RAIDset Information

To view all information about all RAIDsets, type: `HSZ20>` **`show raidsets full`**
The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
----------------------------------------------------------
RAID1         raidset       DISK100       D100
                            DISK110
                            DISK200
     Switches:
        POLICY (for replacement) = BEST_PERFORMANCE
        RECONSTRUCT (Priority) = NORMAL
        CHUNKSIZE = 63 BLOCKS
     State:
        RECONSTRUCT 3% COMPLETE
        DISK100 (MEMBER 0) IS RECONSTRUCTING
        DISK110 (MEMBER 1) IS RECONSTRUCTING
        DISK200 (MEMBER 2) IS RECONSTRUCTING
     Size:  2050353
```

```
RAID2   raidset            DISK120      D200
                           DISK210
                           DISK220
     Switches:
        POLICY (for replacement) = BEST_PERFORMANCE
        RECONSTRUCT (Priority) = NORMAL
        CHUNKSIZE = 63 BLOCKS
     State:
        NORMAL
        DISK120 (MEMBER 0) IS NORMAL
        DISK210 (MEMBER 1) IS NORMAL
        DISK220 (MEMBER 2) IS NORMAL
     Size:  2050353
```

4.2.5.3 Viewing a RAIDset

To view a specific RAIDset, type: `HSZ20>` **`show raid1`**
where RAID1 = the name of a RAIDset
The CLI displays information similar to the following:

```
Name          Storageset    Uses          Used by
---------------------------------------------------------
RAID1         raidset       DISK100       D100
                            DISK110
                            DISK200
     Switches:
        POLICY (for replacement) = BEST_PERFORMANCE
        RECONSTRUCT (Priority) = NORMAL
        CHUNKSIZE = 63 BLOCKS
     State:
        RECONSTRUCT 3% COMPLETE
        DISK100 (MEMBER 0) IS RECONSTRUCTING
        DISK110 (MEMBER 1) IS RECONSTRUCTING
        DISK200 (MEMBER 2) IS RECONSTRUCTING
     Size:  2050353
```

4.2.6 Viewing Units

You can view units in the following ways:

- A list of all units
- A list of all units including their switches, states and sizes

4.2.6.1 Viewing a List of Units

To view a list of units, type: `HSZ20>` **`show units`**

The CLI displays information similar to the following:

```
LUN                               Uses
---------------------------------------
D100                              MIRROR1
```

Table 4–3 describes the information displayed by the show unit command.

Table 4–3 Show Unit Command Display

Show Command Heading	Description
LUN	The LUN number for each device installed in a cabinet slot is 0.
Uses	The name of the device or storageset to which you assigned the specified LUN.

4.2.6.2 Viewing All Unit Information

To view all information about the units, type: `HSZ20>` **`show units full`**
The CLI displays information similar to the following:

```
LUN                                         Uses
-------------------------------------------------
D101                                       RAID1
    Switches:
      RUN   NOWRITE_PROTECT   READ_CACHE
      NOWRITEBACK_CACHE
      MAXIMUM_CACHED_TRANSFER_SIZE = 32
    State:
      ONLINE to this controller
      Not reserved
      PREFERRED_PATH = THIS_CONTROLLER
    Size: 16751956 blocks
```

Table 4–4 describes the information displayed by the show unit full command.

Table 4–4 Show Unit Full Command Display

Information Displayed	Description
Switches	
RUN/NORUN	RUN indicates the host can access the unit. NORUN indicates the host cannot access the unit.
WRITE_PROTECT/ NOWRITE_PROTECT	WRITE_PROTECT indicates data cannot be written to the unit. NOWRITE_PROTECT indicates data can be written to the unit.
READ_CACHE/ NOREAD_CACHE	READ_CACHE indicates that the host reads data from a high speed memory buffer between the unit and the host. NOREAD_CACHE indicates that the host reads data directly from the unit.

Table 4–4 Show Unit Full Command Display (cont.)

Information Displayed	**Description**
Switches	
WRITEBACK_CACHE/ NOWRITEBACK_CACHE	WRITEBACK_CACHE indicates that the cache writes data to the cache memory and flushes the data to the external storage device at a later time. NOWRITEBACK_CACHE indicates the host writes data directly to the unit.
MAXIMUM_CACHE_TRANSFER_SIZE	Indicates how many blocks of data the cache transfers at a given time.
State	
Online to this controller	
Not reserved	
Write-Cache	
Preferred Path = this_controller	
Size	
16751956 Blocks	

4.2.7 Viewing the Spareset

To identify the member devices of the spareset, type: `HSZ20>` **`show spareset`**
The CLI displays information about the spareset similar to the following:

```
Name          Storageset    Uses          Used by
-------------------------------------------------------------
SPARESET      SPARESET      DISK130
```

4.2.8 Viewing the Failedset

To identify the member devices of the failedset, type: `HSZ20>` **`show failedset`**
The CLI displays information about the failedset similar to the following:

```
Name          Storageset    Uses          Used by
-------------------------------------------------------------
FAILEDSET     FAILEDSET     DISK110
```

4.2.9 Viewing Controller Parameters

To view controller and cache information type: `HSZ20>` **`show this_controller`**
The CLI displays information about the controller similar to the following:

```
Controller:
        SC4200 CXS54200234 Firmware V262-0, Hardware A02
        Not configured for dual-redundancy
        SCSI address 7
        Time:  NOT SET
Host Port:
        SCSI address 7
Cache:
        16 megabyte write cache, version 2
        Cache is GOOD
        Battery is GOOD
        No unflushed data in cache
        CACHE_FLUSH_TIMER = FAULT (10 seconds)
        CACHE_POLICY = A
        Host Functionality Mode = A
```

4.3 Viewing Storageset States

Storageset states indicate the operating condition of the storageset based on the operating conditions of each of the member devices of the storageset.

Table 4–5 describes the storageset and member device states.

Table 4-5 Device and Storageset States

State	Device	Storageset
NORMAL	No problems detected.	All members devices are operating under normal conditions.
FAILED	The controller can no longer read or write to the device or has stopped attempting to negotiate with a device after detecting a repetitive or consistent error.	The controller can no longer write to or read from the storageset. All data is lost. Reaching this state varies depending on the type of storageset as follows: Stripeset = one member device has failed. Mirrorset = All members of a mirrorset have failed. Striped Mirrorset = All members of a mirrorset have failed. RAIDset = two members of a RAIDset have failed.

Table 4–5 Device and Storageset States (Cont.)

State	Device	Storageset
REDUCED	Not Applicable.	Reaching this state varies depending on the type of storageset as follows: RAIDset = a single device failed, the RAIDset continues to operate without redundancy. Mirrorset = one or more devices failed depending upon the number of members in the mirrorset. The mirrorset continues to operate normally until only one member remains in the set. When only one member remains in the set, the mirrorset continues to operate but without redundancy. Striped-mirrorset = one or more devices failed. The striped mirrorset continues to operate as long as one member from each mirrorset remains. The mirrorset with only one existing member is no longer redundant.
RECONSTRUCT	The controller is reconstructing the data from a failed device onto this replacement device.	A device failed in a mirrorset, striped-mirrorset, or RAIDset, and the controller has begun to reconstruct the data from the failed device onto a replacement device. If a spareset exists in the subsystem and a replacement policy was set for the storageset, the controller automatically selects a replacement for the failed device from the spareset and begins to reconstruct the failed device's data onto the replacement. If no spareset exists or no replacement policy was set for the storageset, you must remove the failed device from the subsystem cabinet and insert a device in good working order into the same cabinet slot from which you removed the failed device for the storageset to go into a reconstruct state.

You can only view states for storagesets that have been added as a unit. To view storageset states use any of the following commands:

- Show storagesets full
- Show RAID1 (where RAID1 is the name of a specific storageset)
- Show RAIDset full (where RAIDset is a type of storageset)

4.4 Locating Devices Displayed on the CLI in the Subsystem Cabinet

Use the CLI locate command to determine which physical device in the subsystem cabinet corresponds to a device displayed by the CLI. When you issue the locate command, the controller lights the amber fault LED on the corresponding device. You can locate a single device or all the member devices of a storageset.

To locate the physical device that corresponds to a device displayed on the CLI, follow these steps:

1. Type: `HSZ20>`**`locate 1 0 0`** (include a space between each number)
   where: 1 = the device's port address on the controller
   0 = the device's target address on the port
   0 = the device's LUN number on the target
   The controller flashes the amber faultlight of the device at the specified port, target, LUN (ptl) location.
2. Type: `HSZ20>`**`locate cancel`**
   The controller stops flashing the amber fault light on the device.

To locate all the physical devices that are member devices of a storageset, follow these steps:

1. Type: `HSZ20>`**`locate R1`**
   where r1 = the name of a storageset
   The controller flashes the amber fault lights of the member devices of the specified storageset.
2. Type: `HSZ20>`**`locate cancel`**
   The controller stops flashing the amber fault light of the member devices of the specified storageset.

4.5 Recovering After a Device Fails from a Storageset

When a mirrorset or RAIDset looses a member it is in a reduced state. To return to a normal state either:

- The controller automatically adds a new member to the RAIDset from the spareset (if you did not set replacement policy to NOPOLICY when you added the storageset as a unit, and if an appropriate spare is available in the spareset).

  If the controller automatically adds the new device to the storageset, the state of the controller changes to reconstruct and the controller brings the state of the storageset back to normal by reconstructing the failed device's data onto the replacement device. After the controller reconstructs a storageset, you may want to add a new device to replenish the spareset. See Section 4.6 to replace a failed device and then Section 4.6.2 to add a new member device to the spareset.

- You must manually add a new member to the RAIDset.

  If you set the replacement policy to NOPOLICY, see Section 4.6 to replace a failed device and then Section 4.6.1 to replace the failed member of a storageset manually.

4.6 Replacing a Failed Device with a Device in Good Working Order

Replace a failed device with a device in good working order to:

- Return a storageset to a normal state.
  If you set the replacement policy to NOPOLICY, or no spares were available in the spareset, you need to replace the failed device with a device in good working order to return the storageset to a normal state.
- Replenish the spareset.
  Even if you specified a replacement policy, you want to replace the failed device with a device in good working order to replenish the spareset.

The controller puts devices it marks as failed into the failedset.

To remove a device in the failedset from your subsystem, you must delete that device from the failedset.

To replace a failed device with a device in good working order follow these steps:

1. To delete a device from the failed set, use the following CLI command,
   type: `HSZ20>` **`delete failedset disk100`**
   where delete = a CLI command
   failedset = the type of storageset from which you are deleting
   disk100 = the disk which you are deleting
   In addition, you can specify a string of devices, separated by a space.
2. After you delete the device from the failedset, physically remove it from the subsystem.
3. If you want to add a new member to a reduced storageset see Section 4.6.1 Manually Adding a New Device to a Storageset.

   If you want to add a device in good working order as a new member to the spareset, see Section 4.6.2 on adding a device to the spareset.

4.6.1 Manually Adding a New Device to a Storageset

If you set the replacement POLICY parameter for a storageset with a failed device at NOPOLICY, the storageset continues to operate in a reduced state. To return the storageset to a normal state you need to replace the failed member. You can replace the failed member manually or automatically by specifying a replacement policy, adding a device to the spareset and having the controller replace the failed device.

To manually replace a disk member in a reduced storageset with NOPOLICY set for replacement, type: `HSZ20>` **`set R1 replace=disk100`**

where set = a CLI command
r1 = the storageset for which you want to replace a member
disk100 = the name of the replacement disk

The controller adds disk100 to the reduced RAIDset (R1), then begins a reconstruct operation.

NOTE

You cannot specify any other qualifiers with this command.

If you want to replace the failed device automatically, through the spareset, see Section 4.6.2 to add a device to the spareset, and Section 4.7 to select a replacement policy for the storageset.

4.6.2 Adding and Deleting Members to the Spareset

The spareset is a pool of disk drives available to the controller to replace members of a RAIDset or mirrorset that fail. You can add or delete members from the spareset.

To add a device as a member to the spareset, follow these steps:

1. Add the device (if it has not already been added) so that the controller will recognize it. Use the CONFIG utility to automatically add a device, or the CFMENU utility or the ADD DEVICE CLI command to add the device.

> **NOTE**
>
> You do not need to initialize the device. The CLI controller initializes the device when you add it as a member of a spareset.

2. Type: HSZ20> **`add spareset disk100`**
   where add = a CLI command
   spareset = the type of storageset to which you want to add a device
   disk100 = the device that you want to add to the spareset
   You can only specify one device to add to the spareset at a time.

If you want to reconfigure a member device from the spareset, or you want to remove a spareset member device from the subsystem, you must delete it from the spareset.

To delete a device from the spareset, type: `HSZ20>` **`delete spareset disk100`**
where delete = a CLI command
spareset = the type of storage from which you want to delete a member.
disk100 = the member device that you want to remove from the spareset.

4.7 Changing Storageset Parameters Settings

You can change the parameters settings for storagesets that you added as units.

See Table 4–6 to determine what parameters you can change for each type of device and storageset.

Table 4–6 Storageset Parameters

Parameter	**Type of Device/Storageset for Which You Can Change This Parameter**	**Options**
MAXIMUM_CACHED_TRANSFER	Any	1-1024 Blocks 32 = Default
READ_CACHE	Any except RAIDset	READ_CACHE NOREAD_CACHE READ_CACHE=default
RUN	Any	RUN = default NORUN
WRITE_PROTECT	Any	WRITE_PROTECT NOWRITEPROTECT = default
WRITEBACK_CACHE	Any except transportable device	WRITEBACK_CACHE NOWRITEBACK_CACHE = default
POLICY	RAIDset Mirrorset	BEST_FIT BEST_PERFORMANCE = default
RECONSTRUCT	RAIDset Mirrorset	NORMAL = default FAST

For more information about these parameters see Table 4–4, or Chapter 3, Section 3.7 on manually adding devices and storagesets to the host.

To change the settings of storageset parameters, use the set command as follows:
Type: `HSZ20>`**`set d1 write_protect`**
where set = a CLI command
d1 = the unit that contains storageset for which you want to change a parameter setting
write_protect = the new parameter setting

4.8 Removing a Member of a RAIDset

If you want to reconfigure a device currently used as a member of a RAIDset, you can remove it from the RAIDset. If the RAIDset has three members and you remove one, the RAIDset will operate reduced.

> **NOTE**
>
> You cannot remove a member device from a RAIDset if the RAIDset is in a reduced state. In such a case, all data will be lost.
>
> If a replacement policy is specified, the replacement drive is automatically taken from the spareset to replace the removed member using the specified policy.

To remove a member device from a RAIDset, follow these steps:

1. Follow your normal backup procedures to backup the data from the RAIDset prior to removing a member.
2. Type: `HSZ20>` **`set r1 remove=disk100`**

   where set = a CLI command
   r1 = the RAIDset from which you want to remove a member
   disk100 = the name of the member device you want to remove from the RAIDset.

The RAIDset continues to operate reduced until you add a new member. A RAIDset is not redundant in a reduced state.

4.9 Removing a Member Device from a Mirrorset

Since a mirrorset creates two or more devices with exact copies of the same data, you can remove a member device from a mirrorset without losing data.

To remove a member device of a mirrorset, type: `HSZ20>` **`remove disk100`**
where remove = a CLI command
disk100 = the member device that you want to remove from a mirrorset.

You can specify additional devices to remove from the same or other mirrorsets.

> **NOTE**
>
> Use the unmirror command only for mirrorsets with member devices having equal storage capacity sizes.
>
> The capacity of a mirrorset is limited to the size of the mirrorset member with the smallest storage capacity. The portion of a larger member's capacity above the capacity of a smaller member does not get used.
>
> If you unmirror a member with a larger storage capacity than is used in the mirrorset, this would cause a change in the reported capacity size of that device and possibly cause the operating system to not know how to work with it.

4.10 Unmirroring a One Member Mirrorset

If you reduce a mirrorset to one member and then want to use that member device in some other configuration, you can unmirror it.

Only unmirror devices for which you no longer need the data stored on it.

To unmirror a one member mirrorset, type: `HSZ20>` **`unmirror disk100`**

where unmirror = a CLI command
disk100 = a device previously configured into a mirrorset

4.11 Creating Two Stripesets out of a Striped Mirrorset

You can reconfigure a striped mirrorset into two independent stripesets. To do this, you remove a member from each of the mirrorsets that makeup a stripeset. You must remove all of the members from the mirrorsets that you want to use in the new stripeset at the same time.

To reconfigure a striped mirrorset into two independent stripesets, type: `HSZ20>` **`reduce disk100 disk210`**

where reduce = a CLI command
disk100, disk210 = the member devices of the mirrorsets in a striped mirrorset that you want to use for a new stripeset

The controller removes the specified devices from the mirrorsets from which they were members and configures these devices together as a new stripeset.

4.12 Obtaining Snapshot Copies of Data from Mirrorsets

Some system management environments use mirroring as a method to obtain snapshot copies of data for ready backup or to be used to backup to tape. Mirroring is used since member devices of a mirrorset contain identical copies of the same data, and you can add and remove devices from a mirrorset while it is in use.

Some file systems and applications recommend against mirror snapshots as a backup strategy, because the snapshot technique is not well suited to the way they use data. Therefore, check your file system and application documentation for compatibility with this method.

The process for obtaining snapshot copies of data from mirrorsets requires the following general steps:

- Make the copy during normal operation of the mirrorset
- Make the copy by adding a device to the mirrorset from which you want to copy data. As with the addition of any new mirrorset member, the mirroring facility copies the data from the existing members to the new snapshot member.
- When the controller completes copying data to the snapshot member, extract the snapshot member from the mirrorset. (In some situations, you may need to quiesce any applications operating on the mirrorset so that there is no activity on the mirrorset when you extract the snapshot member.)

The snapshot unit will be an identical copy of the original, right down to volume labels and file system information that is normally unique. If you wish to access the snapshot copy on the same system as the original while the original is still active, you need to take the appropriate steps to override the host system's usual protections against such duplication.

You can create a snapshot copy using the CLONE Utility or CLI commands.

4.12.1 Creating a Snapshot Using the CLONE Utility

Using the mirroring facility to create a snapshot copy of host unit data is a multistep process. If your data snapshot needs are straightforward, you can use CLONE to relieve you of much of the chore of performing this common operation. CLONE does nothing that cannot be achieved by issuing the appropriate CLI commands yourself; in fact, CLONE operates by issuing CLI commands for you. These commands are printed on the terminal as part of the program output so that you can see what it has done. This is useful both as a learning tool, and in the event that you need to recover an incomplete CLONE operation manually.

The Clone utility does not allow cloning to a device if the source disk was initialized without the SAVE_CONFIGURATION option and the target device was initialized with the SAVE_CONFIGURATION option. To work around this problem, you can reinitialize the target disk with the NOSAVE_CONFIGURATION.

4.12.1.1 The Cloning Process

Any unit created by CLONE will have a mirrorset level in the configuration hierarchy, even if the original unit did not have mirroring. The CLONE utility uses mirrorsets during copying, and must maintain that structure when it adds the cloned disk as a newly-created unit. The steps below illustrate this concept:

1. CLONE is run on a single-disk unit.
2. CLONE creates a mirrorset from the single disk and adds the target disk to the mirrorset. The target disk is initialized *as a member of a mirrorset* when it is added.
3. The data on the first member of the mirrorset is copied to the new (target) member.
4. When the copy is complete, CLONE removes the target drive from the mirrorset, and then sets the original disk so that it is no longer a mirrorset.
5. Because the target disk was initialized as a mirrorset member, CLONE makes it into a single-member mirrorset to preserve the metadata before adding it as a unit.

The new mirrorset level only affects clones of single-disk units and stripesets. Mirrorsets and striped mirrorsets already use mirroring, and any clones from such units will have the same structure as the original units.

4.12.1.2 Cloning a Single Device Unit

To clone a single device unit, use the follow this example:

```
HSZ20> RUN CLONE

Clone Local Program Invoked
Units available for cloning:
 110
 799
Enter unit to clone ? 110

Clone will create a new unit which is a copy of unit 110.
Enter the unit number which you want assigned to the new unit ?  797
The new unit may be added using one of the following methods:
  1.  Clone will pause after all members have been copied.  The user must
     then press RETURN to cause the new unit to be added.
  2.  After all members have been copied, the unit will be added
automatically.
Under which above method should the new unit be added [] ?  1
Devices available for clone targets:
  DISK300 (size=2050353)
  DISK330 (size=2050353)
Use available device DISK300(size=2050353) for
member DISK110(size=2050353) (y,n) [y] ?  y
       mirror DISK110 C_M
       set C_M nopolicy
       set C_M members=2
       set C_M replace=DISK300
Copy in progress for each new member.  Please be patient...
  copy from DISK110 to DISK300 is 6% complete
  copy from DISK110 to DISK300 is 12% complete
  copy from DISK110 to DISK300 is 19% complete
…
  copy from DISK110 to DISK300 is 98% complete
  copy from DISK110 to DISK300 is 100% complete

Press RETURN when you want the new unit to be created
       reduce  DISK300
       unmirror DISK110
       add mirrorset C_M        DISK300
       init C_M       nodestroy
       add unit D797 C_M

D797 has been created.  It is a clone of D110.

Clone - Normal Termination

HSZ20>
```

4.12.1.3 Cloning a Stripeset

> **NOTE**
>
> CLONE will not operate on stripesets that have different types of members. All stripeset members must be single disks, or all must be mirrorsets.

To clone a stripeset, follow this example:

```
HSZ20> SHOW STORAGESETS

Name          Storageset                      Uses            Used by
----------------------------------------------------------------------------

ST1           stripeset                       DISK130         D799
                                              DISK200
HSZ20> run clone

Clone Local Program Invoked
Units available for cloning:
 110
 799
Enter unit to clone ? 799
Clone will create a new unit which is a copy of unit 799.
Enter the unit number which you want assigned to the new unit ?  798
The new unit may be added using one of the following methods:
  1.  Clone will pause after all members have been copied.  The user must
     then press RETURN to cause the new unit to be added.
  2.  After all members have been copied, the unit will be added
automatically.
Under which above method should the new unit be added [] ?  1

Devices available for clone targets:
  DISK220 (size=832317)
  DISK240 (size=832317)
  DISK310 (size=832317)
Use available device DISK220(size=832317) for
member DISK130(size=832317) (y,n) [y] ? y
        mirror DISK130 C_MA
        set C_MA nopolicy
        set C_MA members=2
        set C_MA replace=DISK220
Devices available for clone targets:
  DISK240 (size=832317)
  DISK310 (size=832317)
Use available device DISK240(size=832317) for
member DISK200(size=832317) (y,n) [y] ? y
        mirror DISK200 C_MB
        set C_MB nopolicy
        set C_MB members=2
        set C_MB replace=DISK240
Copy in progress for each new member.  Please be patient...
  copy from DISK130 to DISK220 is 15% complete
  copy from DISK200 to DISK240 is 11% complete
  copy from DISK130 to DISK220 is 27% complete
  copy from DISK200 to DISK240 is 23% complete
…
  copy from DISK130 to DISK220 is 100% complete
  copy from DISK200 to DISK240 is 100% complete
```

```
Press RETURN when you want the new unit to be created
        reduce  DISK220 DISK240
        unmirror DISK130
        unmirror DISK200
        add mirrorset C_MA       DISK220
        add mirrorset C_MB       DISK240
        add stripeset C_ST1  C_MA C_MB
        init C_ST1     nodestroy chunk=128
        add unit D798 C_ST1

D798 has been created.  It is a clone of D799.

Clone - Normal Termination

HSZ20> SHOW DEVICES

Name          Type          Port Targ  Lun                   Used by
---------------------------------------------------------------------------

DISK130       disk             1    3    0                   ST1
DISK200       disk             2    0    0                   ST1
DISK220       disk             2    2    0                   C_MA
DISK240       disk             2    4    0                   C_MB
DISK310       disk             3    1    0


HSZ20> SHOW STRIPESETS

Name          Storageset                     Uses            Used by
---------------------------------------------------------------------------

C_ST1         stripeset                      C_MA            D798
                                             C_MB

ST1           stripeset                      DISK130         D799
                                             DISK200

HSZ20>
```

4.12.1.4 Cloning a Mirrorset

To clone a mirrorset, follow this example:

```
HSZ20> SHOW DEVICES

Name          Type          Port Targ  Lun                    Used by
----------------------------------------------------------------------------

DISK110       disk             1    1    0                    M1
DISK150       disk             1    5    0
DISK220       disk             2    2    0                    M1
DISK310       disk             3    1    0
DISK350       disk             3    5    0
DISK420       disk             4    2    0
DISK510       disk             5    1    0
DISK550       disk             5    5    0
DISK620       disk             6    2    0

HSZ20>

HSZ20> SHOW MIRRORSETS

Name          Storageset                   Uses            Used by
----------------------------------------------------------------------------

M1            mirrorset                    DISK110         D411
                                           DISK220
HSZ20>

HSZ20> RUN CLONE

Clone Local Program Invoked
Units available for cloning:
 411
Enter unit to clone ?411
Enter the unit number which you want assigned to the new unit? 499
The new unit may be added using one of the following methods:
  1.  Clone will pause after all members have been copied.  The user must
     then press RETURN to cause the new unit to be added.
  2.  After all members have been copied, the unit will be added
automatically.
Under which above method should the new unit be added []? 2
Devices available for clone targets:
  DISK150 (size=832317)
  DISK310 (size=832317)
  DISK350 (size=832317)
  DISK420 (size=832317)
  DISK510 (size=832317)
  DISK550 (size=832317)
  DISK620 (size=832317)
Use available device DISK150(size=832317) for
member DISK110(size=832317) (y,n) [y] ? N
Use available device DISK310(size=832317) for
member DISK110(size=832317) (y,n) [y] ? Y
        set M1 nopolicy
        set M1 members=3
        set M1 replace=DISK310
Copy in progress for each new member.  Please be patient...
  copy from DISK110 to DISK310 is 5% complete
  copy from DISK110 to DISK310 is 10% complete
  …
```

```
 copy from DISK110 to DISK310 is 97% complete
  copy from DISK110 to DISK310 is 100% complete
        reduce DISK310
        add mirrorset C_M1       DISK310
        init C_M1      nodestroy
        add unit D499 C_M1

D499 has been created.  It is a clone of D411.

Clone - Normal Termination

HSZ20>

HSZ20> SHOW DEVICES

Name          Type          Port Targ  Lun                    Used by
----------------------------------------------------------------------------

DISK110       disk             1    1    0                    M1
DISK150       disk             1    5    0
DISK220       disk             2    2    0                    M1
DISK310       disk             3    1    0                    C_M1
DISK350       disk             3    5    0
DISK420       disk             4    2    0
DISK510       disk             5    1    0
DISK550       disk             5    5    0
DISK620       disk             6    2    0

HSZ20>

HSZ20> SHOW MIRRORSETS

Name          Storageset                  Uses             Used by
----------------------------------------------------------------------------

C_M1          mirrorset                   DISK310          D499

M1            mirrorset                   DISK110          D411
                                          DISK220

HSZ20>
```

4.12.1.5 Cloning a Striped Mirrorset

To clone a striped mirrorset, follow this example:

```
HSZ20> SHOW DEVICES

Name          Type          Port Targ  Lun                    Used by
---------------------------------------------------------------------------

DISK110       disk             1    1    0                    M1
DISK150       disk             1    5    0
DISK220       disk             2    2    0                    M1
DISK310       disk             3    1    0                    M2
DISK350       disk             3    5    0
DISK420       disk             4    2    0                    M2
DISK510       disk             5    1    0                    M3
DISK550       disk             5    5    0
DISK620       disk             6    2    0                    M3


HSZ20> SHOW MIRRORSETS

Name          Storageset                    Uses             Used by
---------------------------------------------------------------------------
M1            mirrorset                     DISK110          ST1
                                            DISK220

M2            mirrorset                     DISK310          ST1
                                            DISK420

M3            mirrorset                     DISK510          ST1
                                            DISK620

HSZ20>

HSZ20> SHOW STORAGESETS

Name          Storageset                    Uses             Used by
---------------------------------------------------------------------------

ST1           stripeset                     M1               D411
                                            M2
                                            M3
HSZ20>

HSZ20> RUN CLONE

Clone Local Program Invoked
Units available for cloning:
 411
Enter unit to clone ?411
Clone will create a new unit which is a copy of unit 411.
Enter the unit number which you want assigned to the new unit? 499
The new unit may be added using one of the following methods:
  1.  Clone will pause after all members have been copied.  The user must
     then press RETURN to cause the new unit to be added.
  2.  After all members have been copied, the unit will be added
automatically.
```

```
Under which above method should the new unit be added []? 1
Devices available for clone targets:
  DISK150 (size=832317)
  DISK350 (size=832317)
  DISK550 (size=832317)
Use available device DISK150(size=832317) for
member DISK110(size=832317) (y,n) [y] ? N
Use available device DISK350(size=832317) for
member DISK110(size=832317) (y,n) [y] ? Y
        set M1 nopolicy
        set M1 members=3
        set M1 replace=DISK350
Devices available for clone targets:
  DISK150 (size=832317)
  DISK550 (size=832317)
Use available device DISK150(size=832317) for
member DISK310(size=832317) (y,n) [y] ? Y
        set M2 nopolicy
        set M2 members=3
        set M2 replace=DISK150
Devices available for clone targets:
  DISK550 (size=832317)
Use available device DISK550(size=832317) for
member DISK510(size=832317) (y,n) [y] ? Y
        set M3 nopolicy
        set M3 members=3
        set M3 replace=DISK550
Copy in progress for each new member.  Please be patient...
  copy from DISK110 to DISK350 is 7% complete
  copy from DISK310 to DISK150 is 6% complete
  copy from DISK510 to DISK550 is 5% complete
  …
  copy from DISK110 to DISK350 is 100% complete
  copy from DISK310 to DISK150 is 100% complete
  copy from DISK510 to DISK550 is 100% complete

Press RETURN when you want the new unit to be created
        reduce DISK350 DISK150 DISK550
        add mirrorset C_M1      DISK350
        add mirrorset C_M2      DISK150
        add mirrorset C_M3      DISK550
        add stripeset C_ST1     C_M1 C_M2 C_M3
        init C_ST1    nodestroy chunk=128
        add unit D499 C_ST1

D499 has been created.  It is a clone of D411.

Clone - Normal Termination

HSZ20>

HSZ20> SHOW DEVICES
```

```
Name           Type          Port Targ  Lun                    Used by
---------------------------------------------------------------------------

DISK110        disk             1    1    0                    M1
DISK150        disk             1    5    0                    C_M2
DISK220        disk             2    2    0                    M1
DISK310        disk             3    1    0                    M2
DISK350        disk             3    5    0                    C_M1
DISK420        disk             4    2    0                    M2
DISK510        disk             5    1    0                    M3
DISK550        disk             5    5    0                    C_M3
DISK620        disk             6    2    0                    M3

HSZ20>

HSZ20> SHOW MIRRORSETS

Name           Storageset                     Uses             Used by
---------------------------------------------------------------------------

C_M            mirrorset                      DISK350          C_ST1

C_MA           mirrorset                      DISK150          C_ST1

C_MB           mirrorset                      DISK550          C_ST1

M1             mirrorset                      DISK110          ST1
                                              DISK220

M2             mirrorset                      DISK310          ST1
                                              DISK420

M3             mirrorset                      DISK510          ST1
                                              DISK620

HSZ20>

HSZ20> SHOW STORAGESETS

Name           Storageset                     Uses             Used by
---------------------------------------------------------------------------

C_ST1          stripeset                      C_M1             D499
                                              C_M2
                                              C_M3

ST1            stripeset                      M1               D411
                                              M2
                                              M3
HSZ20>
```

4.12.1.6 Error Recovery from the Clone Utility

Although the operations involved in most CLONE processes are commonplace, the reaction in response to failures during the operation is heavily influenced by each installation's operational needs. For this reason, CLONE does not attempt to automate the error recovery process. If you encounter an error during a CLONE operation, you will need to resolve the situation by manually issuing the appropriate CLI commands.

The following circumstance will cause CLONE to cease operation:

1. If the controller is reset or there is a power loss.
2. If you change the configuration of the unit while CLONE is running.
3. If a disk device being used by CLONE fails.
4. If the CLONE utility is aborted via Ctrl/Y or Ctrl/C.
5. If the controller fails.

In most cases, CLONE can automate the operation entirely and will complete successfully without intervention. In those cases when it cannot complete, it stops at the point the error occurred. The CLI commands performed up to that point remain on the screen.

To recover partially completed CLONE operations, you manually issue the remaining commands in the sequence, or undo the actions CLONE has taken so far by issuing appropriate CLI commands to reverse them.

4.12.2 Creating a Snapshot Using CLI Commands

In addition to the CLONE Utility, you can also create snapshot copies of devices and storagesets using CLI commands.

4.12.2.1 Creating a Snapshot of a Single Device

Following is the list of commands (in sequential order) you would use if you need to manually clone the single device unit:

1. SHOW DEVICES
2. SHOW STORAGESETS
3. MIRROR *disk-device-name1 container-name*
4. SET *mirrorset-container-name* POLICY= *policy-type*
5. SET *mirrorset-container-name* MEMBERSHIP= *number-of-members*
6. SET *mirrorset-container-name* REPLACE= *disk-device-name*
7. Wait for member NORMALIZATION
8. REDUCE *disk-device-name*
9. UNMIRROR *disk-device-name*
10. ADD MIRRORSET *mirrorset-container-name disk-device-name*
11. INITIALIZE *container-name* NODESTROY
12. ADD UNIT *unit-number container-name*

- Following is an example of the specific manual commands used to clone the single device unit:

```
HSZ20> SHOW DEVICES ❶
HSZ20> SHOW STORAGESETS ❷
HSZ20> MIRROR DISK110 C_M ❸
HSZ20> SET C_M NOPOLICY ❹
HSZ20> SET C_M MEMBERSHIP=2 ❺
HSZ20> SET C_M REPLACE=DISK300 ❻
HSZ20> REDUCE DISK300 ❼
HSZ20> UNMIRROR DISK110 ❽
HSZ20> ADD MIRRORSET C_M DISK300 ❾
HSZ20> INITIALIZE C_M NODESTROY ❿
HSZ20> ADD UNIT D100 C_M ⓫
```

❶ Displays the units available for cloning.
❷ Displays the storagesets that are currently configured.
❸ Converts the physical device DISK110 into a one-member mirrorset and names the mirrorset C_M.
❹ Sets the replacement policy for mirrorset C_M to NOPOLICY so that spareset members are not automatically added to the mirrorset
❺ Sets the nominal membership of mirrorset C_M to two members.
❻ Places disk DISK300 into mirrorset C_M. The controller automatically starts copying data from the original member to the new member.
❼ Removes DISK300 from mirrorset C_M, after a copy from the existing mirrorset member to DISK300 has completed.
❽ Converts the one-member mirrorset DISK110 back to a physical device.
❾ Creates a one-member mirrorset from DISK300 and names the mirrorset C_M.
❿ Initializes mirrorset C_M, but does not destroy any of the forced error metadata on the disk.
⓫ Creates a logical unit to the controller from the initialized mirrorset C_M and names the unit.

4.12.2.2 Creating a Snapshot of a Stripeset Using CLI Commands

To manually clone a stripeset, use the following commands in sequential order:

1. SHOW DEVICES
2. SHOW UNITS
3. MIRROR *disk-device-name1 container-name1*
4. SET *mirrorset-container-name1* NOPOLICY
5. SET *mirrorset-container-name1* MEMBERSHIP=
6. SET *mirrorset-container-name1* REPLACE=
7. MIRROR *disk-device-name2 mirrorset-container-name2*
8. SET *mirrorset-container-name2* NOPOLICY
9. SET *mirrorset-container-name2* MEMBERSHIP=
10. SET *mirrorset-container-name2* REPLACE=
    Wait for NORMALIZATION of *all* new mirrorset members.
11. REDUCE *disk-device-name disk-device-name[N]*
12. UNMIRROR *disk-device-name*

13. UNMIRROR *disk-device-name*
14. ADD MIRRORSET *mirrorset-container-name1 disk-device-name*
15. ADD MIRRORSET *mirrorset-container-name2 disk-device-name*
16. ADD STRIPESET *stripeset-container-name mirrorset-container-name1mirrorset-container-name2*
17. INITIALIZE *stripeset-container-name* NODESTROY CHUNK=
18. ADD UNIT *unit-name stripeset-container-name*

Following is an example of the manual commands for cloning a stripeset:

```
HSZ20> SHOW DEVICES ❶
HSZ20> SHOW STORAGESETS ❷
HSZ20> MIRROR DISK130 C_MA ❸
HSZ20> SET C_MA NOPOLICY ❹
HSZ20> SET C_MA MEMBERSHIP=2 ❺
HSZ20> SET C_MA REPLACE=DISK220 ❻
HSZ20> MIRROR DISK200 C_MB ❼
HSZ20> SET C_MB NOPOLICY ❽
HSZ20> SET C_MB MEMBERSHIP=2 ❾
HSZ20> SET C_MB REPLACE=DISK240 ❿

...wait for NORMALIZATION OF DISK220 and DISK240...

HSZ20> REDUCE DISK220 DISK240 ⓫
HSZ20> UNMIRROR DISK130 ⓬
HSZ20> UNMIRROR DISK200 ⓭
HSZ20> ADD MIRRORSET C_MA DISK220 ⓮
HSZ20> ADD MIRRORSET C_MB DISK240 ⓯
HSZ20> ADD STRIPESET C_ST1 C_MA C_MB ⓰
HSZ20> INITIALIZE C_ST1 NODESTROY CHUNK=128 ⓱
HSZ20> ADD UNIT D C_S⓲
HSZ20> SHOW DEVICES ⓳
HSZ20> SHOW STRIPESETS ⓴
```

❶ Displays the units available for cloning.
❷ Displays the storagesets that are currently configured.
❸ Converts the physical device DISK130 into a one-member mirrorset and names the mirrorset C_MA.
❹ Sets the replacement policy for mirrorset C_MA to NOPOLICY so that spareset members are not automatically added to the mirrorset
❺ Sets the nominal membership of mirrorset C_MA to two members.
❻ Places disk DISK220 into mirrorset C_MA. The controller automatically starts copying data from the original member to the new member.
❼ Converts the physical device DISK200 into a one-member mirrorset and names the mirrorset C_MB.
❽ Sets the replacement policy for mirrorset C_MB to NOPOLICY so that spareset members are not automatically added to the mirrorset
❾ Sets the nominal membership of mirrorset C_MB to two members.
❿ Places disk DISK240 into mirrorset C_MB. The controller automatically starts copying data from the original member to the new member.
⓫ Removes DISK220 and DISK240 from their respective mirrorsets, after the copy operations from the existing mirrorset members has completed.
⓬ Converts the one-member mirrorset DISK130 back to a physical device.

⓭ Converts the one-member mirrorset DISK200 back to a physical device.
⓮ Creates a one-member mirrorset from DISK220 and names the mirrorset C_MA.
⓯ Creates a one-member mirrorset from DISK240 and names the mirrorset C_MB.
⓰ Creates a stripeset called C_ST1 from the two newly-created mirrorsets C_MA and C_MB.
⓱ Initializes strip eset C_ST1 with a chunksize of 128 blocks, but does not write new metadata onto the members.
⓲ Creates a logical unit for the host from the initialized stripeset C_ST1 and names the unit.
⓳ Shows the device configuration after the clone is complete.
⓴ Shows the stripesets after the clone is complete.

4.12.2.3 Creating a Snapshot of a Mirrorset Using CLI Commands

To manually clone a mirrorset, use the following commands in sequential order:

1. SHOW DEVICES
2. SHOW MIRRORSETS
3. SET *mirrorset-container-name* NOPOLICY
4. SET *mirrorset-container-name* MEMBERSHIP=3
5. SET *mirrorset-container-name* REPLACE= *disk-device-name*
   Wait for NORMALIZATION.
6. REDUCE *disk-device-name*
7. ADD MIRRORSET *mirrorset-container-name disk-device-name*
8. INITIALIZE *mirrorset-container-name* NODESTROY CHUNK=
9. ADD UNIT *unit-name mirrorset-container-name*
10. SHOW DEVICES
11. SHOW MIRRORSETS

Following is an example of the manual commands for cloning a mirrorset:

```
HSZ20> SHOW DEVICES ❶
HSZ20> SHOW MIRRORSETS ❷
HSZ20> SET M1 NOPOLICY ❸
HSZ20> SET M1 MEMBERSHIP=3 ❹
HSZ20> SET M1 REPLACE=DISK310 ❺

...wait for NORMALIZATION of M1...

HSZ20> REDUCE DISK310 ❻
HSZ20> ADD MIRRORSET C_M1 DISK310❼
HSZ20> INITIALIZE C_M1  NODESTROY CHUNK=128❽
HSZ20> ADD UNIT D C_M1 ❾
HSZ20> SHOW DEVICES ❿
HSZ20> SHOW MIRRORSETS ⓫
```

❶ Displays the units available for cloning.
❷ Displays the mirrorsets that are currently configured.
❸ Sets the replacement policy for mirrorset M1 to NOPOLICY so that spareset members are
not automatically added to the mirrorset
❹ Sets the nominal membership of mirrorset M1 to three members.
❺ Places disk DISK310 into mirrorset M1. The controller automatically starts copying data to the new member to make it identical to the existing NORMAL members.
❻ Removes DISK310 from mirrorset M1 after the copy operation has completed.
❼ Creates a mirrorset called C_M1 from DISK310.
❽ Initializes mirrorset C_M1 with a chunksize of 128 blocks, but does not write new metadata onto the members.
❾ Creates a logical unit for the host from mirrorset C_M1 and names the unit.
❿ Shows the device configuration after the clone is complete.
⓫ Shows the mirrorsets after the clone is complete.

4.12.2.4 Creating a Snapshot of a Striped Mirrorset Using CLI Commands

To manually clone a striped mirrorset, use the following commands in sequential order:

1. SHOW DEVICES
2. SHOW MIRRORSETS
3. SHOW STORAGESETS
4. SET *mirrorset-container-name1* NOPOLICY
5. SET *mirrorset-container-name1* MEMBERSHIP=
6. SET *mirrorset-container-name1* REPLACE=
7. SET *mirrorset-container-name2* NOPOLICY
8. SET *mirrorset-container-name2* MEMBERSHIP=
9. SET *mirrorset-container-name2* REPLACE=
10. SET *mirrorset-container-name3* NOPOLICY
11. SET *mirrorset-container-name3* MEMBERSHIP=
12. SET *mirrorset-container-name3* REPLACE=
    Wait for NORMALIZATION of *all* new mirrorset members.
13. REDUCE *disk-device-name disk-device-name[N] disk-device-name [N]*
14. ADD MIRRORSET *mirrorset-container-name disk-device-name*
15. ADD MIRRORSET *mirrorset-container-name disk-device-name*
16. ADD MIRRORSET *mirrorset-container-name disk-device-name*
17. ADD STRIPESET *stripeset-name mirrorset-container-name1 mirrorset-container-name2 mirrorset-container-name3*
18. INITIALIZE *stripeset-name* NODESTROY CHUNK=
19. ADD UNIT *unit-name stripeset-name*
20. SHOW DEVICES

21. SHOW MIRRORSETS
22. SHOW STORAGESETS

Following is an example of the manual commands for cloning a striped mirrorset:

```
HSZ20> SHOW DEVICES ❶
HSZ20> SHOW MIRRORSETS ❷
HSZ20> SHOW STORAGESETS ❸
HSZ20> SET M1 NOPOLICY ❹
HSZ20> SET M1 MEMBERSHIP=3 ❺
HSZ20> SET M1 REPLACE=DISK350 ❻
HSZ20> SET M2 NOPOLICY ❼
HSZ20> SET M2 MEMBERSHIP=3 ❽
HSZ20> SET M2 REPLACE=DISK150 ❾
HSZ20> SET M3 NOPOLICY ❿
HSZ20> SET M3 MEMBERSHIP=3 ⓫
HSZ20> SET M3 REPLACE=DISK550 ⓬

...wait for NORMALIZATION of M1, M2, and M3...

HSZ20> REDUCE DISK350 DISK150 DISK550⓭
HSZ20> ADD MIRRORSET C_M1  DISK350⓮
HSZ20> ADD MIRRORSET C_M2  DISK150⓯
HSZ20> ADD MIRRORSET C_M3  DISK550⓰
HSZ20> ADD STRIPESET C_ST1 C_M1 C_M2 C_M3⓱
HSZ20> INITIALIZE C_ST1 NODESTROY⓲
HSZ20> ADD UNIT D499 C_ST1·⓳
HSZ20> SHOW DEVICES ⓴
HSZ20> SHOW MIRRORSETS ㉑
HSZ20> SHOW STORAGESETS ㉒
```

❶ Displays the devices available for cloning.
❷ Displays the configured mirrorsets.
❸ Displays the configured storagesets.
❹ Sets the replacement policy for mirrorset M1 to NOPOLICY so that spareset members are not automatically added to the mirrorset
❺ Sets the nominal membership of mirrorset M1 to three members.
❻ Places disk DISK350 into mirrorset M1. The controller automatically starts copying data from existing NORMAL members to the new member.
❼ Sets the replacement policy for mirrorset M2 to NOPOLICY so that spareset members are not automatically added to the mirrorset
❽ Sets the nominal membership of mirrorset M2 to three members.
❾ Places disk DISK150 into mirrorset M2. The controller automatically starts copying data from existing NORMAL members to the new member.
❿ Sets the replacement policy for mirrorset M3 to NOPOLICY so that spareset members are not automatically added to the mirrorset
⓫ Sets the nominal membership of mirrorset M3 to three members.
⓬ Places disk DISK550 into mirrorset M3. The controller automatically starts copying data from existing NORMAL members to the new member.
⓭ Removes DISK350, DISK150, and DISK550 from their respective mirrorsets, after the copy operations from the existing NORMAL mirrorset members has completed.
⓮ Creates new one-member mirrorset C_M1 that contains DISK350.
⓯ Creates new one-member mirrorset C_M2 that contains DISK150.
⓰ Creates new one-member mirrorset C_M3 that contains DISK550.

⓱ Creates a stripeset called C_ST1 from the three new mirrorsets C_M1, C_M2, and C_M3.
⓲ Initializes stripeset C_ST1 with a chunksize of 128 blocks, but does not write new metadata onto the members.
⓳ Creates a logical unit for the host from stripeset C_ST1 and names the unit D499.
⓴ Shows the device configuration after the clone is complete.
㉑ Shows the mirrorsets after the clone is complete.
㉒ Shows the storagesets after the clone is complete

4.13 Modifying the Controller Firmware/NVMEM

Use the Code Load/Code Patch (CLCP) utility to modify the controller in the following ways:

- Code Loading - allows you to upgrade the firmware in the controller through the maintenance port.
- Code Patching - allows you to alter memory locations controller's Non Volatile Memory (NVMEM) while the controller is active.

4.13.1 Determining the Firmware Revision Level

To determine the current firmware revision level of the HSZ20 controller in your subsystem, type the following: `HSZ20>`**`show this_controller`**.
The firmware displays information about the controller including the firmware revision level as *`V27Z`*.

4.13.2 Invoking the CLCP utility

Invoke the CLCP:

- On a maintenance or virtual host terminal to perform a code patch
- On a CPU to perform a code load (download firmware).

Use the following CLI command to invoke the CLCP utility, type:

```
HSZ20> run clcp
```

The utility displays a message similar to the following:

```
Select an option from the following list:

Code Load & Code Patch local program Main Menu

 0: Exit
 1: Enter Code LOAD local program
 2: Enter Code PATCH local program

 Enter option number  (0..2) [0] ?
```

4.13.3 Upgrading the Controller Firmware

To upgrade controller firmware, you copy a new image of the firmware from the diskette on which the new image was distributed, to a hard drive and then to the controller. To do this, you need the following:

- A CPU with a floppy diskette drive in which you can insert the diskette with the new image of the firmware
- A CPU with a hard drive with a minimum of 2.5 MB of free disk space
- A terminal program that runs the Kermit protocol. (If operating in an MS-Windows environment, you must use an MS-DOS based terminal program.)

In addition, prior to invoking the CLCP, stop any terminate-and-stay-ready (TSR) programs, such as screen savers, that may be running on that computer.

Set the terminal program to run as follows:

- Kermit transfer protocol
- 19,200 baud
- 8 data bits
- no parity
- one stop bit

NOTE

You must set the RAID Array 310 controller to the same baud rate as the terminal program. To set the controller baud rate to 19,200, type: `HSZ20>`**`Set this_controller terminal speed=19200`**.

It takes approximately 27 minutes to download the new firmware image.

To upgrade controller firmware, follow these steps:

1. Insert the distribution diskette containing the binary file of the new firmware image into a floppy diskette drive of a computer connected to the RAID Array 310 controller.
2. Copy the firmware from the diskette onto a hard drive.
3. Click on the firmware .EXE file to extract the firmware .IMG file.
4. Invoke the CLCP utility as described in Section 4.13.1.

5. Type: **`1`** and press the Return key. The utility displays the following message:

```
You have selected the Code Load local program.  This
program is used to load a new firmware image into the
program card currently inserted in the controller.

Type ^Y or ^C (then RETURN) at any time to abort code
load.

The code image may be loaded using SCSI Write Buffer
commands through the SCSI Host Port, or using KERMIT
through the Maintenance Terminal Port. Enter Y (then
RETURN) to use the SCSI Host Port [Y]: ?

WARNING: proceeding with Code Load will overwrite the
         current content of your program card with a new
         image. Enter Y (then RETURN) to continue [N]: ?
```

6. Exit any MS-Windows sessions and use the KERMIT file transfer protocol to download the new binary firmware. After the download completes, the utility displays the following message:

```
KERMIT file transferred successfully.

Program card is being re-programmed with new file.

*** Do not interrupt this step ***

 Manufacturer code read from memory card= 8989

 Device Code read from memory card= bdbd
```

7. If the main menu appears, press 0 to exit the utility.
8. Verify that this process successfully copied the new version of the firmware onto the controller. At the CLI prompt, type: `HSZ20>` **`show this_controller`**. The CLI displays information indicating the current firmware version of the controller

4.13.4 Entering Code Patches

1. Obtain the appropriate patch data for your controller's firmware version from your Digital representative.
2. Invoke the CLCP program.
3. Type: 2 and press the Return key. The utility displays the following message:

```
You have selected the Code Patch local program.  This
program is used to manage firmware code patches.  Select
an option from the following list:

Type ^Y or ^C (then RETURN) at any time to abort Code
Patch.


Code Patch Main Menu

0: Exit

1: Enter a Patch

2: Delete Patches

3: List Patches

Enter option number (0..3) [0] ?
```

The following sections describe each of the code patch options in detail.

> **NOTE**
>
> The patch data in these examples is provided only for the purposes of illustrating the code patch operation. Obtain actual code patch data for your controller's firmware from your service provider.

Consider the following when using the code patch utility:

- The controller reserves enough nonvolatile memory for approximately ten (10) patches. However, this number varies according to the size of the patches you enter.
- Each patch is associated with only one firmware version, and the Code Patch program verifies the patch against the currently installed firmware.
- Patches are hierarchical. In other words, patch number one (1) must be entered before you enter patch number two (2), and so on. Furthermore, there are no "0" patches. Patches are always numbered sequentially beginning with "1."
- Because of the hierarchical patch structure, removing any patch also removes all higher numbered patches. For example, deleting patch number two (2) also removes patches three (3), four (4), and so on.

4.13.4.1 Enter a Patch Option

This option enables you to enter a firmware program patch directly into the controller's RAM. You are prompted to enter the firmware version number to which the patch applies, the patch length, the patch type, the patch number, the count, the RAM address, the new contents of that address, and a patch verification number.

The code patch utility verifies that the patch you are entering is appropriate for the firmware version in the controller, and that there are no required dependent patches. It allows you to enter only one patch at a time. The utility prompts with error messages if you attempt to perform an illegal patch entry. Following is an example of the use of the patch entry option:

```
HSZ20>     RUN CLCP
Select an option from the following list:
Code Load & Code Patch local program Main Menu
0: Exit
1: Code LOAD
2: Code PATCH
Enter option number (0..2) [0] ? 2
--------------------------------------------------------
You have selected the Code Patch local program.  This program
is used to manage firmware code patches.  Select an option
from the following list:
Type ^Y or ^C (then RETURN) at any time to abort Code Patch.
```

```
Code Patch Main Menu
   0: Exit
   1: Enter a Patch
   2: Delete Patches
   3: List Patches
Enter option number (0..3) [0] ? 1
This is the Enter a Code Patch option.  The program prompts
you  for the patch information, one line at time.  Be careful
to enter the information exactly as it appears on the patch
release. Patches may be installed for any version of
firmware; however, patches entered for firmware versions
other than V27Z   are not applied until the matching version
of firmware is installed.
To enter any patch, you must first install all patches with
lower patch numbers than the patch you are entering,
beginning with patch number 1, for a specific firmware
version. If you incorrectly enter the patch information, you
are given the option to review the patch one line at a time.
Type ^Y or ^C (then RETURN) at any time to abort Code Patch.
Do you wish to continue (y/n) [y] ? Y
Version:       ? V27Z
Length:        ? 10
Patch Type:    ? 0
Patch Number: ? 1
Count:         ?  1
Address:       ? 00000099
Value[  0]     ? 00000000
Count:         ? 0
Verification: ? fdd6e08f
The patch you just entered, is not applied until the
controller is restarted.
Code Patch Main Menu
   0: Exit
   1: Enter a Patch
   2: Delete Patches
   3: List Patches
Enter option number (0..3) [0] ?
CLCP - Normal Termination
Restart of the controller required to apply new patch.
   HSZ20> restart this_controller
```

4.13.4.2 Delete Patches Option

The Delete Patches option enables you to remove previously-installed patches from controller RAM. The program displays the currently-installed patches and patches to be deleted.

The code patch utility verifies that the patch requested for deletion exists, and that it is not a dependent patch for a higher-numbered installed patch. It allows you to delete only one patch at a time. The utility prompts with error messages if you attempt to perform an illegal patch deletion.

Following is an example of the use of the patch deletion option.

```
   HSZ20>     RUN CLCP
   Select an option from the following list:
   Code Load & Code Patch local program Main Menu
   0: Exit
   1: Code LOAD
   2: Code PATCH
   Enter option number (0..2) [0] ?  2
   --------------------------------------------------------
You have selected the Code Patch local program.  This program
is used to manage firmware code patches.  Select an option
from the following list:
Type ^Y or ^C (then RETURN) at any time to abort Code Patch.
Code Patch Main Menu
   0: Exit
   1: Enter a Patch
   2: Delete Patches
   3: List Patches
Enter option number (0..3) [0] ?  2
This is the Delete Patches option.  The program prompts you
for the firmware version and patch number you wish to delete.
If you select a patch for deletion that is required for
another patch, all dependent patches are also selected for
deletion.  The program lists your deletion selections and
asks if you wish to continue.
Type ^Y or ^C (then RETURN) at any time to abort Code Patch.
The following patches are currently stored in the patch area:
 Firmware Version - Patch number(s)
     ______   _________
     V27Z    -        1
Currently, 97% of the patch area is free.

Firmware Version of patch to delete ?  V27Z
Patch Number to delete ?  1
```

```
The following patches have been selected for deletion:
 Firmware Version - Patch number(s)
     ______   _________
     V27Z    -        1
Do you wish to continue (y/n) [y] ?   Y
The patch you have just deleted is currently applied, but
will not be applied when the controller is restarted.
Code Patch Main Menu
   0: Exit
   1: Enter a Patch
   2: Delete Patches
   3: List Patches
Enter option number (0..3) [0] ?
```

4.13.4.3 List Patches Option

The List Patches option enables you to display a listing of controller firmware versions, and the currently-installed patches that apply to them.

Following is an example of the use of the patch listing option:

```
   HSZ20>     RUN CLCP
   Select an option from the following list:
   Code Load & Code Patch local program Main Menu
   0: Exit
   1: Code LOAD
   2: Code PATCH
   Enter option number (0..2) [0] ? 2
   --------------------------------------------------------
You have selected the Code Patch local program.  This program
is used to manage firmware code patches.  Select an option
from the following list:
Type ^Y or ^C (then RETURN) at any time to abort Code Patch.
Code Patch Main Menu
   0: Exit
   1: Enter a Patch
   2: Delete Patches
   3: List Patches
Enter option number (0..3) [0] ? 3
The following patches are currently stored in the patch area:
 Firmware Version - Patch number(s)
    ______    _________
    V27Z -    1
    V27Z -    1
Currently, 94% of the patch area is free.
Code Patch Main Menu
   0: Exit
   1: Enter a Patch
   2: Delete Patches
   3: List Patches
Enter option number (0..3) [0] ?
```

NOTE

The SHOW *controller* command also provides patch information in the form of a "dash number" following the firmware version. In the following example, firmware version 2.7 has had patches applied up to patch number three (3):

```
HSZ20> SHOW THIS_CONTROLLER
Controller:
     HSZ20 ZG33400026 Firmware V27Z-3, Hardware A-02
   .
   .
   .
```

4.13.4.4 Responding to Code Patching Messages

All patching must be exact or the firmware image in controller NVMEM does not operate. For this reason, the Code Patch utility does not allow you to incorrectly enter or delete patch information at any time.

In these cases the program provides messages to assist you with understanding any problems and corrective actions. The messages appear during interactive use of Code Patch (rather than, for example, at the HSZ20 prompt) as each condition arises. Following are messages you may encounter while using Code Patch to enter and delete patches.

```
Firmware Version x does not have any patches to delete.
```

Explanation: You cannot delete a patch because the firmware version entered does not have any patches entered.

```
Firmware Version x does not have patch number x to delete.
```

Explanation: You cannot delete this patch because the firmware version entered does not have the specified patch entered.

```
The patch you entered is already installed on this
controller.
```

Explanation: The specified patch is already present in the patch area of controller memory. If you wish to reenter this patch, first use the Delete Patch option.

```
The patch you are entering requires other patches to be
entered.
```

Explanation: You have attempted to enter a patch without first entering the lower numbered patches in the hierarchy. Enter all patches for this firmware version that have lower numbers than the current patch. Then enter the current patch.

```
WARNING The patch you are entering is not for the current
firmware version x.
```

Explanation: The patch you are entering applies to a firmware version other than the one currently installed in the controller. Code Patch allows you to enter the patch; however, the patch is not applied until its correct firmware version is installed.

```
You incorrectly entered the patch information.
```

Explanation: The patch information was not entered exactly. The program prompts you for each line of the patch entry, with the default from your previous response. Verify that each entry is exactly the same as the patch release. If you choose not to continue, or if you abort during this review procedure, the patch information you entered is lost and you must enter the entire patch again. You may enter Ctrl/Z <Return> at any prompt to choose the default for the remaining entries.

```
The patch you have just entered is not applied until the
controller firmware is changed to Version x.
```

Explanation: The patch entered applies to a firmware version other than the one currently installed in the controller. Code Patch does not apply the patch until its correct firmware version is installed.

```
You have requested deletion of a patch number that another
patch requires.
```

Explanation: You are attempting to delete a patch in the hierarchy that has higher numbered patches entered. Code Patch allows you to proceed; however, the program deletes all the higher numbered patches in the hierarchy (for this firmware version) along with the specified patch.

4.13.4.5 Exiting Code Patch

Exit Code Patch by choosing option 0 from the main menu. (Pressing Ctrl/C or Ctrl/Y at any time during Code Patch also will abort and exit the program.)

4.14 Setting Feature Licenses

Use the firmware licensing system (FLS) to enable or disable the licensed value-added software features (RAID, mirroring, and write-back cache) for the RAID Array 310. Use the FLS utility to do the following:

- Enable or disable optional functions for your controller
- Try an optional feature before purchasing the license to use it
- Change your license key for an option

Start FLS from the CLI prompt. After starting, the FLS display shows the current status of the value-added options for your controller and contains menu choices for each function of the utility.

4.14.1 Enabling Options

You can turn on any option at any time with FLS, but if you enable an option for which you are not licensed, an error message appears on your CLI console and an error is logged in the host error log. These error indications are repeated at least once each hour while the unlicensed option remains enabled.

4.14.2 Disabling Options

You cannot disable an option if that option is currently in use. Table 4–7 lists the conditions under which you can disable an FLS option.

Table 4–7 Disabling FLS Options

Option	Conditions Required to Disable
RAID	No RAIDset configured
WBCA	Write-back caching not in use on any unit
MIRR	No mirrorset configured

4.14.3 License Key

When you first run FLS, the license key is cleared. If you obtain a license for a firmware option, you will receive a customer license key. This key contains two parts: a customer identification string from 6 to 32 characters long, and an 8-character **cyclic redundancy check** (CRC) string. You must enter the customer identification string with the CRC string appended to it when you use FLS.

4.14.4 Using the Menu

You can perform the operations listed in Table 4–8 from the FLS menu:

Table 4–8 FLS Operations

Action	Submenu Choices	Result
Enable an option	List each option and its status	Selection enabled
Disable an option	List each option and its status	Selection disabled
Enter a license key	Prompt for new license key	Entered key checked for validity
Clear a license key	Prompt for license key to clear	Entered key becomes invalid

4.14.5 Examples

The following example shows the FLS main menu:

```
HSZ20> RUN FLS

------------------------------------------------------------------
      Firmware Licensing System (FLS) on node BERT
Option❶   State❷       License❸       Key❹
------------------------------------------------------------------
 RAID      DISABLED      INVALID       *none*
 WBCA      ENABLED ******INVALID!****** *none*
 MIRR      ENABLED        VALID        ACME_WIDGET_CORP........

                RAID = RAID Option❺
                WBCA = Writeback Cache Option
                MIRR = Disk Mirroring Option
------------------------------------------------------------------
    1.  Enable a firmware option
    2.  Disable a firmware option
    3.  Enter a license key for a firmware option
    4.  Clear a license key for a firmware option
    0.  Exit FLS
Enter selection (0:4) [0] ?
```

❶ **Option**–The RAID, write-back cache (WBCA), and mirror (MIRR) options are available.

❷ **State**–Both MIRR and WBCA are enabled. You may use any option that is enabled, regardless of whether you have a valid license key.

❸ **License**–WBCA is running without a valid license. This status will show when you are running an option on a trial basis. The license becomes valid when you enter a license key that FLS verifies as valid. You receive this key when you obtain a software option.

❹ **Key+CRC**–The license key is ACME_WIDGET_CORP; the 8-character CRC portion of the key is shown as hidden text (........).

❺ **Description of Option**–A short description of each option is given.

To perform an operation, enter the choice number and any information requested by the submenu or prompts. The following example demonstrates how to enter a license key and enable write-back caching.

```
HSZ20> RUN FLS

------------------------------------------------------------------
      Firmware Licensing System (FLS) on node MASS
Option      State        License        Key
------------------------------------------------------------------
 RAID      DISABLED      INVALID       *none*
 WBCA      DISABLED      INVALID       *none*
 MIRR      DISABLED      INVALID       *none*

                RAID = RAID Option
                WBCA = Writeback Cache Option
                MIRR = Disk Mirroring Option
------------------------------------------------------------------

    1.  Enable a firmware option
    2.  Disable a firmware option
    3.  Enter a license key for a firmware option
    4.  Clear a license key for a firmware option
    0.  Exit FLS
Enter selection (0:4) [0] ? 3
```

```
    1.  Enter new license key+CRC for RAID   (current key is invalid)
    2.  Enter new license key+CRC for WBCA   (current key is invalid)
    3.  Enter new license key+CRC for MIRR   (current key is invalid)
    0.  Return to main menu
Enter selection (0:3) [0] ? 2❶
Enter new WBCA key, including 8-character CRC, or enter 0
        to return to main menu: ACME_WIDGET_CORPVB8UWQ9C ❷(2)

*** License key verified ***


------------------------------------------------------------------
      Firmware Licensing System (FLS) on node MASS
Option      State        License        Key
------------------------------------------------------------------
 RAID      DISABLED      INVALID        *none*
 WBCA      DISABLED       VALID         ACME_WIDGET_CORP........
 MIRR      DISABLED      INVALID        *none*

                RAID = RAID Option
                WBCA = Writeback Cache Option
                MIRR = Disk Mirroring Option
------------------------------------------------------------------
    1.  Enable a firmware option
    2.  Disable a firmware option
    3.  Enter a license key for a firmware option
    4.  Clear a license key for a firmware option
    0.  Exit FLS
Enter selection (0:4) [0] ? 1

    1.  Enable RAID
    2.  Enable WBCA
    3.  Enable MIRR
    0.  Return to main menu
Enter selection (0:3) [0] ? 2❸
*** WBCA enabled ***


------------------------------------------------------------------
      Firmware Licensing System (FLS) on node MASS
Option      State        License        Key
------------------------------------------------------------------
 RAID      DISABLED      INVALID        *none*
 WBCA      ENABLED        VALID❹(4)       ACME_WIDGET_CORP........
 MIRR      DISABLED      INVALID        *none*

                RAID = RAID Option
                WBCA = Writeback Cache Option
                MIRR = Disk Mirroring Option
------------------------------------------------------------------
    1.  Enable a firmware option
    2.  Disable a firmware option
    3.  Enter a license key for a firmware option
    4.  Clear a license key for a firmware option
    0.  Exit FLS
```

❶ The user chooses to enter a new license key for WBCA.

❷ The user enters the new license key, along with the customer license key, which is displayed as it is entered.

❸ The user enables write-back cache.

❹ This entry in the FLS display shows that write-back cache is enabled under a valid license.

4.14.6 Messages

This section lists the messages that you may receive from FLS.

Message:

`option has been turned on without a valid license`

Explanation: You have activated the option named by *option* without entering a valid license key. You can evaluate this option for a time to determine its value, and you will receive a valid license key when you purchase the license for the option.

Message:

`Error nnnn: option support is not enabled on this controller`

Explanation: The *option* you are attempting to use is not enabled in FLS. For example, if you try to turn on write-back caching, you will receive this error if write-back caching is not enabled by FLS.

Message:

`***WARNING:  This is an invalid license Key+CRC***`

Explanation: The license key you entered is not between 6 and 32 characters, or the customer license key is not valid. Verify that the key is correct and reenter.

Message:

```
***Error: Disabling option is not possible at this time,
option is in use ***
```

Explanation: You have attempted to disable the option named by *option* while it is in use. Refer to Section 4.14.2 for more information on disabling options.

4.15 Diagnosing Cache Battery State

The onboard controller firmware checks the condition of the writeback cache batteries every 24 hours. If the cache batteries fail the firmware diagnostic, the firmware makes the RAIDsets and mirrorsets inoperative.

In addition, use the following commands to view information about the cache and battery state, even when the cache is disabled:

```
show this_controller
show this_controller full
```

4.16 Replacing a Failed Controller

Follow the procedure described in the *RAID Array 310 Deskside Subsystem User's Guide* EK-SMCPL-UG to physically remove a failed controller and replace it with a controller in good working order.

> **NOTE**
>
> Do not change any devices in your subsystem while replacing a controller. Adding or removing devices may interfere with the controller's ability to restore the most current copy of the configuration.

If the new controller you install already contained some configuration information, it may not restore and restart. In the event that the new controller does not start, use the following proceudre to delete any existing configuration on the new controller and restart it.

1. Type: `HSZ20>` **`set this_controller initial_configuration`**
   This command deletes any existing configuration on the controller.
   The controller halts after completing this command.
2. Press the reset button on the controller to start the controller. The controller scans the devices in the subsystem, loads the latest copy of the configuration and then restarts.
3. Connect the host bus cable.

5

Monitoring Subsystem Performance

This chapter describes the FMU and DILX utility local diagnostic programs, which you can use to view error logs and other information about the performance of the subsystem, and to analyze the performance of the subsystem by exercising it.

Use the Fault Management Utility (FMU), VTDPY Utility and the DILX local diagnostic programs to monitor subsystem performance by viewing error logs and exercising the subsystem.

5.1 Viewing Error Logs

Use the Fault Management Utility to do the following:

- Control the spontaneous event logging and last failure logging displays
- Display controller last failure and memory system failure information
- Review some event log information during a terminal session.

FMU will only interpret errors that occur after you install and run controller firmware containing FMU. In other words, you cannot install FMU after an error occurs to use FMU to troubleshoot the error.

After invoking FMU, you may perform the functions described in the following sections. Defaults are specified by "D."

NOTE

Some examples in this chapter may include references to tape, tape loader, and CDROM devices; however, the RAID Array 310 controller does not support such devices.

5.1.1 Invoking the FMU Utility

To invoke and Run the FMU utility interactively during a terminal session, type:

```
HSZ20> RUN FMU
```

5.1.2 Using the FMU SET Commands

The FMU SET command enables certain functions and parameters in the FMU utility, and controls the content of displays. The SET command works with two spontaneous displays:

- Event logging (EVL)
- Last failure logging (LFL)

and also controls the options for the interactive displays available under the SHOW command.

5.1.2.1 Setting EVENT_Logging

SET EVENT_LOGGING
SET NOEVENT_LOGGING (D)

This command enables/disables the event log display on the maintenance terminal. With the event log display enabled, the controller spontaneously displays EIP information during your terminal session. The first line of an event log display begins with "%EVL."

Event log displays are inhibited during the execution of both CLI commands and utilities invoked from a maintenance terminal. Events that are reported while a maintenance terminal is in use will not appear when the terminal again becomes available. (The %EVL display will be lost.)

> **NOTE**
>
> Execution of a CLI command or utility does not begin until you press Return. If FMU reports an event during command line input before Return is pressed, the %EVL display will interrupt the input.
>
> Following the %EVL display, the CLI prompt and command input entered prior to the interruption will be redisplayed. You can then complete the current command line (unless FMU reports another event).

5.1.2.2 Setting Last_Failure_Logging

SET LAST_FAILURE_LOGGING
SET NOLAST_FAILURE_LOGGING (D)

This command enables/disables the last failure log display on the maintenance terminal. With the last failure log display enabled, the controller spontaneously displays information relevant to the sudden termination of executing firmware. The first line of a last failure log display begins with "%LFL."

In cases where an automatic hardware reset occurs (such as power failure, pressing the reset button, and so on) the last failure log display is inhibited because automatic reset does not allow sufficient time to complete the display.

5.1.2.3 Setting Logging Repair_Action_Disiplay

SET EVENT_LOGGING REPAIR_ACTION_DISPLAY
SET EVENT_LOGGING NOREPAIR_ACTION_DISPLAY (D)
SET LAST_FAILURE_LOGGING REPAIR_ACTION_DISPLAY
SET LAST_FAILURE_LOGGING NOREPAIR_ACTION_DISPLAY (D)

This command and qualifier enables/disables recommended repair action display for event logging and last failure logging displays. With recommended repair action display enabled, the controller displays all of the recommended repair actions associated with the Instance Code and/or Last Failure Code used to describe an event.

5.1.2.4 Setting Logging Verbose

SET EVENT_LOGGING VERBOSE
SET EVENT_LOGGING NOVERBOSE (D)
SET LAST_FAILURE_LOGGING VERBOSE
SET LAST_FAILURE_LOGGING NOVERBOSE (D)

This command and qualifier enables/disables descriptive text for event logging and last failure logging displays.

The display always identifies the various fields and their numeric content that comprise an event/last failure log. With verbosity enabled, the controller also displays a description of the numeric value in each log field if appropriate.

5.1.2.5 Setting Prompt_Display

SET PROMPT_DISPLAY
SET NOPROMPT_DISPLAY (D)

This command enables/disables the CLI prompt string display within the first line of event logging and last failure logging displays, as shown in the following example (using %EVL):

%EVL-- Instance Code: 01010302
%EVL--HSZ20> -- Instance Code: 01010302

5.1.2.6 Setting Timestamp_Display

SET TIMESTAMP_DISPLAY
SET NOTIMESTAMP_DISPLAY (D)

This command enables/disables current timestamp string display within the first line of event logging and last failure logging displays, as shown in the following example (using %EVL):

```
%EVL-- Instance Code: 01010302
%EVL--07-JUL-1994 07:44:48-- Instance Code: 01010302
```

You can use combinations of the SET [NO]PROMPT_DISPLAY and SET [NO]TIMESTAMP_DISPLAY commands to provide the following types of event logging and last failure logging first line displays (examples using %EVL):

```
%EVL--HSZ20> --07-JUL-1994 07:44:48-- Instance Code: 01010302
%EVL--07-JUL-1994 07:44:48-- Instance Code: 01010302
%EVL--HSZ20> Instance Code: 01010302
%EVL-- Instance Code: 01010302
```

The last example shown is recommended when VCS is in use, since the controller identification (prompt string) and timestamp information is already supplied by VCS.

5.1.2.7 Setting Repair_Action_Display

SET FMU REPAIR_ACTION_DISPLAY
SET FMU NOREPAIR_ACTION_DISPLAY (D)

This command enables/disables the recommended repair action display for FMU SHOW LAST_FAILURE and SHOW MEMORY_SYSTEM_FAILURE command output. With recommended repair action display enabled, the command output displays all of the recommended repair actions associated with the Instance Code and/or Last Failure Code used to describe an event.

5.1.2.8 Setting FMU Verbose

SET FMU VERBOSE
SET FMU NOVERBOSE (D)

This command enables/disables descriptive text for FMU SHOW LAST_FAILURE and SHOW MEMORY_SYSTEM_FAILURE command output. The output always identifies the various fields and their numeric content that comprise an event/last failure log. With verbosity enabled, the controller also displays a description of the numeric value in each log field if appropriate.

5.1.2.9 Setting Permanent

SET EVENT_LOGGING [qualifier ... qualifier] PERMANENT
SET NOEVENT_LOGGING PERMANENT
SET LAST_FAILURE_LOGGING [qualifier ... qualifier] PERMANENT
SET NOLAST_FAILURE_LOGGING PERMANENT
SET FMU [qualifier ... qualifier] PERMANENT

The PERMANENT qualifier stores the parameter setting specified by the primary keyword and optional qualifiers in nonvolatile memory so that the setting is preserved across controller resets. In addition, when PERMANENT is specified, the given setting takes effect immediately.

If the PERMANENT qualifier is not specified, the given setting takes effect immediately. However, it remains in effect only as long as the current FMU session remains active or until the setting is changed by a subsequent SET command.

When running FMU from a maintenance terminal, changing EVENT_LOGGING parameters without specifying the PERMANENT qualifier has no effect. However, the same action while running FMU from a virtual terminal is effective. This permits the EVENT_LOGGING operation to be changed on a temporary basis only from a virtual terminal.

You can specify multiple additional qualifiers on the same command line for the SET EVENT_LOGGING, SET LAST_FAILURE_LOGGING, and SET FMU commands. For example, the following are all valid commands:

```
FMU> SET EVENT_LOGGING PERMANENT
FMU> SET LAST_FAILURE_LOGGING NOREPAIR_ACTION_DISPLAY PERMANENT
FMU> SET EVENT_LOGGING REPAIR_ACTION_DISPLAY NOVERBOSE PERMANENT
```

5.1.3 Using FMU SHOW Commands

The SHOW command controls the interactive reviewing of last failure and memory system failure information.

5.1.3.1 Showing Last_Failure

SHOW LAST_FAILURE qualifier [additional qualifier]

This command interactively displays the last failure information stored in nonvolatile memory. Information related to the most recent and three previous last failure events is stored in a separate entry in nonvolatile memory. Table 5−1 describes the SHOW LAST_FAILURE Command Qualifiers.

Table 5−1 SHOW LAST_FAILURE Command Qualifiers and Additional Qualifiers

Qualifier	Description
MOST_RECENT	Displays the most recent last failure information.
ALL	Displays the last four failure event entries. Events are displayed in descending order, starting with the most recent.
ENTRY n	Displays one of the last four entries. You must supply an entry number (range 1–4).
Additional Qualifier	**Description**
FULL	When included on the same command line with the MOST_RECENT, ALL, or ENTRY qualifier, displays extended information valuable to Digital Multivendor Customer Services.

5.1.3.2 Showing Memory_System_Failure

SHOW MEMORY_SYSTEM_FAILURE qualifier

This command interactively displays memory system failure information from any of the last failure entries stored in nonvolatile memory. Table 5−2 describes the SHOW MEMORY_SYSTEM_FAILURE command qualifiers.

Table 5−2 Show Memory_System_Failure Command Qualifiers

Qualifier	Description
MOST_RECENT	Displays the most recent memory system failure information contained in any of last failure information entries. (Note that the most recent memory system failure may not be the most recent last failure.)
ALL	Displays all memory system failure information contained in any or all of the four last failure information entries, in most recent to least recent order.
ENTRY n	Displays memory system failure information contained in one last failure information entry (range 1–4).

NOTE

The FULL additional qualifier is not available with the SHOW MEMORY_SYSTEM_FAILURE command. To obtain the extended information associated with the selected memory system failure, perform a SHOW LAST_FAILURE ENTRY n FULL, where n is the last failure entry number identified in the memory system failure display.

The following message appears when FMU cannot access error information for the SHOW command:

```
(***Last Failure Entry x EDC bad; translation terminated***)
```

5.1.3.3 SHOWING Parameters

SHOW PARAMETERS

This command displays the current/permanent setting of parameters affected by the SET command.

5.1.3.4 SHOWING Time

SHOW TIME

This command displays the current controller time (i.e., wall clock time) and the current controller power on time in the following format:

```
Time: 07-FEB-1995 10:14:20
Power On Time: 0.Years, 0. Days, 8. Hours, 27. Minutes, 8.Seconds
```

If the controller time has not yet been set when the show times command is issued, the format looks like this:

```
Time: 07-FEB-1995 10:14:20 (time not set)
Power On Time: 0.Years, 0. Days, 8. Hours, 27. Minutes, 8.Seconds
```

5.1.4 Using DESCRIBE Commands

The describe command displays descriptive text for one or more numeric values contained in a particular event log field.

DESCRIBE ASC_ASCQ_CODE
DESCRIBE COMPONENT_CODE
DESCRIBE CONTROLLER_UNIQUE_ASC_ASCQ_CODE
DESCRIBE DEVICE_TYPE_CODE
DESCRIBE EVENT_THRESHOLD_CODE
DESCRIBE INSTANCE_CODE
DESCRIBE LAST_FAILURE_CODE
DESCRIBE REPAIR_ACTION_CODE
DESCRIBE RESTART_TYPE
DESCRIBE SCSI_COMMAND_OPERATION_CODE
DESCRIBE SENSE_DATA_QUALIFIERS
DESCRIBE SENSE_KEY_CODE
DESCRIBE TEMPLATE_CODE

All DESCRIBE qualifiers require at least one numeric value parameter. DESCRIBE qualifiers requiring multiple numeric value parameters are footnoted as such.

Type a question mark (?) in place of a numeric value parameter in order to identify the value and range required, as shown in the following example. Note that when sequential values are required, you must supply values for the earlier parameters before entering a question mark for the later parameter in the sequence.

```
FMU> DESCRIBE ASC_ASCQ_CODE ?
Your options are:
   ASC value (range: 0 through FF hexadecimal)
FMU> DESCRIBE ASC_ASCQ_CODE 0 ?
Your options are:
   ASCQ value (range: 0 through FF hexadecimal)
FMU> DESCRIBE ASC_ASCQ_CODE 0 0 ?
Your options are:
   SCSI Device Type value (range: 0 through FF hexadecimal)
FMU>
```

5.1.5 Exiting FMU

The EXIT command terminates FMU and returns you to the CLI prompt. (You also may enter Ctrl/C or Ctrl/Y to abort FMU.)

5.1.6 Understanding FMU Displays

This section presents examples that show some of the output information available when using FMU. Values enclosed in parentheses are hexadecimal translations of decimal numbers.

```
FMU> SHOW LAST_FAILURE ENTRY 4

Last Failure Entry: 4[1] Flags: 0007FA80[2]

%FMU-01-Last Failure Event, Instance Code: 01010302[3]

 Power On Time: 0 Years, 41 Days, 4 Hours, 49 Minutes, 8 Seconds[4]
 [5]Controller Model: SC4600 Serial Number: ZG30355555 Hardware
Version: 0000(00)
 Controller Identifier:
  Unique Device Number: 000130355555 Model: 40(28) Class: 1(01)
 HSOF Version: V20(20)[6]
 Node Name: "CLI307" CI Node Number: 7(07)[7]
 Informational Report
 [2]Instance Code 01010302 Description:
  An unrecoverable hardware detected fault occurred.
 [9]Last Failure Code: 018800A0 (No Last Failure Parameters)
 Last Failure Code 018800A0 Description:
  A processor interrupt was generated with an indication that the
program card was removed.

FMU> SHOW MEMORY_SYSTEM_FAILURE ENTRY 2

Last Failure Entry: 2[1] Flags: 0007FA8C[2]
 Memory System Failure indicated

%FMU-14-Memory System Failure Event, Instance Code: 016E2D02[3]
 Power On Time: 0 Years, 41 Days, 21 Hours, 5 Minutes, 39 Seconds[4]
 [5]Controller Model: SC4600 Serial Number: ZG30355555 Hardware
Version: 0000(00)
 Controller Identifier:
  Unique Device Number: 000130355555 Model: 40(28) Class: 1(01)
 HSOF Version: V20(20)[6]
 Node Name: "HSJ307" CI Node Number: 7(07)[7]
 Reported via non-maskable interrupt
 [8]Memory Address: 40000000
```

```
 Byte Count: 0(00000000)
 DRAB Registers:
  DSR:  2D17403F  CSR:  8000A220 DCSR:  00003403  DER:  00001C00
EAR:  04000000
  EDR:  F4000003  ERR:  00000000  RSR:  09805432  CHC:  E7FFFFFC
CMC:  90A5FEF0
 Diagnostic Registers:
  RDR0: E7FFFFFC  RDR1: 90A5FEF0  WDR0: 7F021000  WDR1: FF06020D
 9Instance Code 016E2D02 Description:
  The CACHEA0 DRAB detected a Nonexistent Memory Error condition
during an I960 attempt to read CACHEA0 memory.
```

[1] **Entry** – A number representing the last failure entry position.
[2] **Flags** – This value should be recorded and reported to Digital Multivendor Customer Services.
[3] **Instance Code** – The instance code (and description) associated with this failure.
[4] **Power On Time** – The time of failure.
[5] **Controller** – Information identifying your controller.
[6] **HSOF Version** – Firmware version.
[7] **Node Name** – Node identifier.
[8] **Last Failure Code** – Last failure code and description.
[9] **Memory Address** – The memory address, byte count, and register contents should be recorded and reported to Digital Multivendor Customer Services.

FMU> **SHOW LAST_FAILURE ENTRY 4 FULL**

```
Last Failure Entry: 4 Flags: 0007FA80

%FMU-01-Last Failure Event, Instance Code: 01010302
 Power On Time: 0 Years, 41 Days, 4 Hours, 49 Minutes, 8 Seconds
 Controller Model: SC4600 Serial Number: ZG30355555 Hardware Version: 0000(00)
 Controller Identifier:
  Unique Device Number: 000130355555 Model: 40(28) Class: 1(01)
 HSOF Version: V20(20)
 Node Name: "HSJ307" CI Node Number: 7(07)
 Informational Report
 Instance Code 01010302 Description:
  An unrecoverable hardware detected fault occurred.
 Last Failure Code: 018800A0 (No Last Failure Parameters)
 Last Failure Code 018800A0 Description:
  A processor interrupt was generated with an indication that the program
  card was removed.
(1)Current Thread: NULL Current I960 Priority: 001F0000
 Interrupt Stack Guard is intact
 Thread Stack Guard State Flags (ID# Bit; 0=intact,1=not intact): 00000000
 I960 Stack:
  Levels: 2
  Level 0:
   Return type: Interrupt
   PFP: 201C15F7  SP:  201FABD0  RIP: 200CF898  R3:  201C15F7
   R4:  00000000  R5:  00000000  R6:  00000000  R7:  00000000
   R8:  00000000  R9:  00000000  R10: 00000000  R11: 00000000
   R12: 00000000  R13: 00000000  R14: 00000000  R15: D87FA8FE
  Level 1:
   Return type: Local
   PFP: 00000000  SP:  201C1680  RIP: 200D0AC0  R3:  00000000
   R4:  00000000  R5:  00000000  R6:  00000000  R7:  00000000
   R8:  00000000  R9:  00000000  R10: 00000000  R11: 00000000
   R12: 00000000  R13: 00000000  R14: 00000000  R15: 00000000
   G0:  00000000  G1:  00000000  G2:  2011DFF4  G3:  00000000
   G4:  00000000  G5:  00400000  G6:  201148B0  G7:  00000003
   G8:  000000CC  G9:  00000001  G10: 00000004  G11: 2011F108
   G12: 200E969C  G13: 201D3D8C  G14: 00000000  FP:  201FAB50
```

```
 Diagnostic Registers:
  RDR0: E7FFFFFC  RDR1: E7FFFEF0  WDR0: 7F021000  WDR1: FF06020D
 Master DRAB Registers:
  DSR:  2D170CBC  CSR:  00000000 DCSR:  0009FFFF  DER:  00001C00  EAR:
00200400
  EDR:  00000000  ERR:  00000023  RSR:  00801432  CHC:  005BB41A  CMC:
002E853E
CACHEA0 DRAB Registers:
  DSR:  2D17003F  CSR:  00000000 DCSR:  00003403  DER:  00001C76  EAR:
00000000
  EDR:  FFFFFFFF  ERR:  00000000  RSR:  09805432  CHC:  00000000  CMC:
00000000
 CACHEA1 DRAB Registers:
  Not Available.
 CACHEB0 DRAB Registers:
  DSR:  2C17003D  CSR:  00000000 DCSR:  00003402  DER:  00001C76  EAR:
00000000
  EDR:  FFFFFFFF  ERR:  00000000  RSR:  09805432  CHC:  00000000  CMC:
00000000
 CACHEB1 DRAB Registers:
  Not Available.
 FX Registers:
  PCX[0]:   00000000  PCX[1]:   00000001  PCX[2]:   00000201  PCX[3]:
00000401
  PCX[4]:   00000601  PCX[5]:   00000801  PCX[6]:   00000A01  PCX[7]:
00000000
  CSR:      0087E002  GEN_PCX:  00000000  UNUSED0:  00000000  UNUSED1:
00000000
  DILP:     00000000  DADDR:    00000000  DCMD:     00000000
 Host Port Registers (YACI):
  SET:    072E004A  CIA:    F8070700  PCS:    0505400F  RTS:    00000000
  RADILP: 201BEE08  RBDILP: 201BEE2C  TADILP: 201BEE50  TBDILP: 201BEE74
  RADFPA: 2063E760  RBDFPA: 20619560  TADFPA: 20637FA0  TBDFPA: 2062D5A0
  RADNPA: 2063E760  RBDNPA: 20619560  TADNPA: 20637FA0  TBDNPA: 2062D5A0
  REV:    00000003  DIAG:   00000000
 Device Port 0 Registers (NCR710):
  SCNTL0:  CA  SCNTL1:  20  SDID:    00  SIEN:    AF  SCID:    80  SXFER:   00
  SODL:    00  SOCL:    00  SFBR:    00  SIDL:    00  SBDL:    00  SBCL:    00
  DSTAT:   90  SSTAT0:  00  SSTAT1:  00  SSTAT2:  00  DSA:     00000000
  CTEST0:  70  CTEST1:  F0  CTEST2:  21  CTEST3:  08  CTEST4:  00  CTEST5:  00
  CTEST6:  FC  CTEST7:  80  TEMP:    00000000         DFIFO:   00  ISTAT:   00
  CTEST8:  21  LCRC:    00  DBC:     000000           DCMD:    54
  DNAD:    20578F50         DSP:     20578F50         DSPS:    0000015C
  SCRATCH: 0000000          DMODE:   80  DIEN:    27  DWT:     4E  DCNTL:   21
  ADDER:   205790AC Device Port 1 Registers (NCR710):
  SCNTL0:  DA  SCNTL1:  20  SDID:    08  SIEN:    AF  SCID:    80  SXFER:   18
  SODL:    41  SOCL:    00  SFBR:    00  SIDL:    00  SBDL:    00  SBCL:    00
  DSTAT:   90  SSTAT0:  00  SSTAT1:  00  SSTAT2:  0F  DSA:     3E000000
  CTEST0:  70  CTEST1:  F0  CTEST2:  25  CTEST3:  41  CTEST4:  00  CTEST5:  00
  CTEST6:  B0  CTEST7:  80  TEMP:    205788B0         DFIFO:   00  ISTAT:   00
  CTEST8:  21  LCRC:    08  DBC:     000000           DCMD:    54
  DNAD:    205788B8         DSP:     205788B8         DSPS:    0000015C
  SCRATCH: 205788B0         DMODE:   80  DIEN:    27  DWT:     4E  DCNTL:   21
  ADDER:   20578A14
 Device Port 2 Registers (NCR710):
  SCNTL0:  CA  SCNTL1:  20  SDID:    00  SIEN:    AF  SCID:    80  SXFER:   00
  SODL:    00  SOCL:    00  SFBR:    00  SIDL:    00  SBDL:    00  SBCL:    00
  DSTAT:   90  SSTAT0:  00  SSTAT1:  00  SSTAT2:  00  DSA:     00000000
  CTEST0:  70  CTEST1:  F0  CTEST2:  21  CTEST3:  08  CTEST4:  00  CTEST5:  00
  CTEST6:  FC  CTEST7:  80  TEMP:    00000000         DFIFO:   00  ISTAT:   00
  CTEST8:  21  LCRC:    00  DBC:     000000           DCMD:    54
  DNAD:    20578220         DSP:     20578220         DSPS:    0000015C
  SCRATCH: 00000000         DMODE:   80  DIEN:    27  DWT:     4E  DCNTL:   21
  ADDER:   2057837C
```

```
Device Port 3 Registers (NCR710):
  SCNTL0:  DA  SCNTL1:  20  SDID:    01  SIEN:    AF  SCID:    80  SXFER:   18
  SODL:    02  SOCL:    00  SFBR:    00  SIDL:    00  SBDL:    00  SBCL:    00
  DSTAT:   90  SSTAT0:  00  SSTAT1:  00  SSTAT2:  0F  DSA:     3E000000
  CTEST0:  70  CTEST1:  F0  CTEST2:  25  CTEST3:  02  CTEST4:  00  CTEST5:  00
  CTEST6:  80  CTEST7:  80  TEMP:    20577B80         DFIFO:   00  ISTAT:   00
  CTEST8:  21  LCRC:    01  DBC:     000000           DCMD:    54
  DNAD:    20577B88         DSP:     20577B88         DSPS:    0000015C
  SCRATCH: 20577B80         DMODE:   80  DIEN:    27  DWT:     4E  DCNTL:   21
  ADDER:   20577CE4
 Device Port 4 Registers (NCR710):
  SCNTL0:  DA  SCNTL1:  20  SDID:    01  SIEN:    AF  SCID:    80  SXFER:   18
  SODL:    42  SOCL:    00  SFBR:    00  SIDL:    00  SBDL:    00  SBCL:    00
  DSTAT:   90  SSTAT0:  00  SSTAT1:  00  SSTAT2:  0F  DSA:     3E000000
  CTEST0:  70  CTEST1:  F0  CTEST2:  25  CTEST3:  42  CTEST4:  00  CTEST5:  00
  CTEST6:  E8  CTEST7:  80  TEMP:    205774E8         DFIFO:   00  ISTAT:   00
  CTEST8:  21  LCRC:    01  DBC:     000000           DCMD:    54
  DNAD:    205774F0         DSP:     205774F0         DSPS:    0000015C
  SCRATCH: 205774E8         DMODE:   80  DIEN:    27  DWT:     4E  DCNTL:   21
  ADDER:   2057764C
 Device Port 5 Registers (NCR710):
  SCNTL0:  DA  SCNTL1:  20  SDID:    02  SIEN:    AF  SCID:    80  SXFER:   68
  SODL:    F6  SOCL:    00  SFBR:    00  SIDL:    00  SBDL:    00  SBCL:    00
  DSTAT:   90  SSTAT0:  00  SSTAT1:  00  SSTAT2:  0F  DSA:     3E000000
  CTEST0:  70  CTEST1:  F0  CTEST2:  25  CTEST3:  80  CTEST4:  00  CTEST5:  00
  CTEST6:  50  CTEST7:  80  TEMP:    20576E50         DFIFO:   00  ISTAT:   00
  CTEST8:  21  LCRC:    20  DBC:     000000           DCMD:    54
  DNAD:    20576E58         DSP:     20576E58         DSPS:    0000015C
  SCRATCH: 20576E50         DMODE:   80  DIEN:    27  DWT:     4E  DCNTL:   21
  ADDER:   20576FB4
```

❶ Information presented after this callout is extended information available with the FULL option. The output shows register contents for various hardware as well as other data. You should print a copy of this display so that Digital Multivendor Customer Services has more information with which to troubleshoot the controller.

5.2 Viewing Subsystem Performance Information with the VTDPY Utility

The VTDPY utility gathers and displays system state and performance information for the controller. The information displayed includes processor utilization, host port activity and status, device state, logical unit state, and cache and I/O performance.

The VTDPY utility requires a video terminal that supports ANSI control sequences, such as a VT220, VT320, or VT420 terminal. A graphics display that provides emulation of an ANSI compatible video terminal can also be used. VTDPY can be run only on terminals connected to the maintenance terminal port. It cannot be run using a virtual host maintenance terminal.

The following sections show how to use the VTDPY utility.

5.2.1 Running VTDPY

You can run only one VTDPY session on each controller at one time. Prior to running VTDPY, set the terminal to NOWRAP mode to prevent the top line of the display from scrolling off of the screen.

To initiate VTDPY from a maintenance terminal at the HSZ20> prompt, enter the following command:

```
HSZ20> run vtdpy
```

5.2.1.1 Using the VTDPY Control Keys

Table 5–3 lists the VTDPY display control key sequences.

Table 5–3 VTDPY Display Control Key Sequences

Control Key Sequence	Function
Ctrl/C	Prompts for commands.
Ctrl/G	Updates the screen (same as Ctrl/Z).
Ctrl/O	Pauses or resumes screen updates.
Ctrl/R	Refreshes current screen display (same as Ctrl/W).
Ctrl/W	Refreshes current screen display (same as Ctrl/R).
Ctrl/Y	Terminates VTDPY and resets screen characteristics.
Ctrl/Z	Updates the screen (same as Ctrl/G).

5.2.1.2 Using the VTDPY Command Line

VTDPY contains a command line interpreter that you can invoke by entering Ctrl/C any time after starting the program. The command line interpreter is used to modify the characteristics of the VTDPY display. Commands also exist to duplicate the function of the control keys listed in Section 5.2.1.1. Table 5–4 displays the VTDPY commands.

Table 5–4 VTDPY Commands

Command String	Function
DISPLAY CACHE	Use 132 column unit caching statistics display.
DISPLAY DEFAULT	Use default 132 column system performance display.
DISPLAY DEVICE	Use 132 column device performance display.
DISPLAY STATUS	Use 80 column controller status display.
EXIT	Terminates program (same as QUIT).
INTERVAL <seconds>	Changes update interval.
HELP	Displays help message text.
REFRESH	Refreshes the current display.
QUIT	Terminates program (same as EXIT).
UPDATE	Updates screen display.

The keywords in the command strings can be abbreviated to the minimum number of characters that are necessary to uniquely identify the keyword. Entering a question mark (?) after a keyword causes the parser to provide a list of keywords or values that can follow the supplied keyword. The command line interpreter is not case sensitive, so keywords can be entered in uppercase, lowercase, or mixed case.

Upon successful execution of a command other than HELP, the command line interpreter is exited and the display is resumed. Entering a carriage return without a command also exits the command line interpreter and resumes the display. If an error occurs in the command, the user prompts for command expansion help, or the HELP command is entered, the command line interpreter prompts for an additional command instead of returning to the display.

5.2.2 Interpreting the VTDPY Display Fields

This section describes the major fields in the VTDPY displays. Figures 5−1 through 5−4 show examples of the VTDPY screens. An explanation of each field of the screens follows the figures.

Figure 5–1 VTDPY Default Display for the Controller

```
SC4200 S/N:  CX12345678 SW: V20Z HW: 00-00 VTDPY Monitor
                                         61.4% Idle     927 KB/S            300 Rq/S
   Pr      Name    Stk/Ma   Typ    Sta    CPU%  SCSI                   Unit     ASWC     KB/S   Rd   Wr   C  H
                     x                          Target 2                                        %    %    m  t
                                                                                                          %  %
0           NULL   0/   0          Rn     61.4                         D0000     o^b      112   62   37   0  0
2          RECON   10/  1   FNC    B1      0.0                         D0001     o^b      118   61   38   0  0
3           SHIS   40/  7   FNC    Rn     32.3  Xfer Rate              D0002     o^b      122   67   32   0  0
8          VTDPY   10/  3   DUP    Rn      0.2  Id   Mnz               D0003     o^b      128   65   34   0  0
18        SCSIVT   10/  1   FNC    B1      0.0  0=Asynch               D0004     o^b       93   69   30   0  0
19         DS_HB   10/  1   FNC    B1      0.0  1=Asynch       Target  D0005     o^b      112   67   32   0  0
24            VA   10/  1   FNC    B1      0.0  2=This       01234567  D0006     o^b      156   66   33   0  0
25          DS_1   40/  6   FNC    B1      5.5  3=Asynch     P1D D D  H D0007    o^b       83   71   28   0  0
26          DS_0   20/  1   FNC    B1      0.4  4=Asynch     o2 D D DDH
27       CLIMAIN   16/  7   FNC    B1      0.0  5=Asynch     r3D D D  H
28         NVFOC   10/  1   FNC    B1      0.0  6=3.57       t4 D D  DH
29        REMOTE   10/  1   FNC    B1      0.0  7=Asynch      5D D D  H
30           FOC   20/  2   FNC    B1      0.0                6 D D DDH
31         DUART   10/  1   FNC    B1      0.0

                                                                                       Up:  0  1:32.46
                                                Ht%    Unit    ASWC    KB/S    Rd%    Wr%    Cm%    HT%
                                                 0
                                                 0
                                                 0
                                                 0
                                                 0
                                                 0
                                                 0
                                                 0
```

Figure 5–2 VTDPY Device Performance Display

```
SC4200 S/N:  CX12345678 SW: V20Z HW: 00-00 VTDPY Monitor
                                                     0.0% Idle          2389 KB/S          348 Rq/S
  Target                                            PTL   ASW   Rq/   RdK   WrKB/  Qu   Tg   CR   BR   T
                                                          F     S     B/S   S      e                   R
   01234567                                         D100  A^     0     0      0    0    0    0    0   0
 P1DDFDDDhH                                         D200  A^    57     0    396   16   11    0    0   0
 o2DDDDDDhH                                         D300  A^     0     0      0    0    0    0    0   0
 r3DDDDDDhH                                         D400  A^     0     0      0    0    0    0    0   0
 t4DDDDDDhH                                         D500  A^     0     0      0    0    0    0    0   0
  5DDDDDDhH                                         D600  A^     0     0      0    0    0    0    0   0
  6DDDDDDhH                                         D110  A^     0     0      0    0    0    0    0   0
                                                    D210  A^    50     0    340   17   11    0    0   0
                                                    D310  A^     0     0      0    0    0    0    0   0
                                                    D410  A^     0     0      0    0    0    0    0   0
                                                    D510  A^     0     0      0    0    0    0    0   0
                                                    D610  A^     0     0      0    0    0    0    0   0
Por   Rq/    RdKB/S   WrKB/S   CR   BR   TR         F120    v F  0     0      0    0    0    0    0   0
 t     S
1      0       0        0      0    0    0          D220  A^    51     0    333   16   11    0    0   0
2    348       1     2387      0    0    0          D320  A^     0     0      0    0    0    0    0   0
3      0       0        0      0    0    0          D420  A^     0     0      0    0    0    0    0   0
4      0       0        0      0    0    0          D520  A^     0     0      0    0    0    0    0   0
5      0       0        0      0    0    0          D620  A^     0     0      0    0    0    0    0   0
6      0       0        0      0    0    0          D130  A^     0     0      0    0    0    0    0   0
                                                    D230  A^    83     0    608   16   11    0    0   0
                                                    D330  A^     0     0      0    0    0    0    0   0

                                                                                 03-FEB-1996  16:53:06
                                                                                       Up:    0  0:25:26
                                                    TR   PTL   ASWF   Rq/S   RdKB/S   WrKB/S   Que   Tg   CR   BR   TR
                                                    0    D430  A^      0       0        0       0    0    0    0    0
                                                    0    D530  A^      0       0        0       0    0    0    0    0
                                                    0    D630  A^      0       0        0       0    0    0    0    0
                                                    0    D140  A^      0       0        0       0    0    0    0    0
                                                    0    D240  A^     60       0      415      16   11    0    0    0
                                                    0    D340  A^      0       0        0       0    0    0    0    0
                                                    0    D440  A^      0       0        0       0    0    0    0    0
                                                    0    D540  A^      0       0        0       0    0    0    0    0
                                                    0    D640  A^      0       0        0       0    0    0    0    0
                                                    0    D150  A^      0       0        0       0    0    0    0    0
                                                    0    D250  A^     45       0      292      16   11    0    0    0
                                                    0    D350  A^      0       0        0       0    0    0    0    0
                                                    0    D450  A^      0       0        0       0    0    0    0    0
                                                    0    D550  A^      0       0        0       0    0    0    0    0
                                                    0    D650  A^      0       0        0       0    0    0    0    0
                                                    0
                                                    0
                                                    0
                                                    0
                                                    0
                                                    0
```

•

Figure 5–3 VTDPY Unit Cache Performance Display

```
SC4200 S/N: CX12345678 SW: V20Z HW: 00-00 VTDPY Monitor
                                           80.1% Idle             2719 KB/S
Unit    ASWE   KB/S    Rd%    Wr%   Cm%   HT%   PH%   MS%   Purge   BlChd   BlHit  Unit    ASWE
D0410   a^ r      0      0      0     0     0     0     0       0       0       0  D0444   a^ r
D0411   a^ r      0      0      0     0     0     0     0       0       0       0  D0445   a^ r
D0413   a^ r      0      0      0     0     0     0     0       0       0       0  D0450   a^ r
D0414   a^ r      0      0      0     0     0     0     0       0       0       0  D0451   a^ r
D0415   a^ r      0      0      0     0     0     0     0       0       0       0  D0452   a^ r
D0420   o^ r    483      0    100     0     0     0   100       0     132       0  D0453   a^ r
D0421   o^ r    483      0    100     0     0     0   100       0     132       0  D0454   a^ r
D0422   o^ r    476      0    100     0     0     0   100       0     132       0  D0455   a^ r
D0423   o^ r    315      0     99     0     1     0    99       0     159       0  D0460   a^ r
D0424   o^ r    483      0    100     0     0     0   100       0     132       0  D0461   a^ r
D0425   o^ r    476      0    100     0     0     0   100       0     132       0  D0462   a^ r
D0430   a^ r      0      0      0     0     0     0     0       0       0       0  D0463   a^ r
D0431   a^ r      0      0      0     0     0     0     0       0       0       0  D0464   a^ r
D0432   a^ r      0      0      0     0     0     0     0       0       0       0  D0465   a^ r
D0433   a^ r      0      0      0     0     0     0     0       0       0       0
D0434   a^ r      0      0      0     0     0     0     0       0       0       0
D0435   a^ r      0      0      0     0     0     0     0       0       0       0
D0440   a^ r      0      0      0     0     0     0     0       0       0       0
D0441   a^ r      0      0      0     0     0     0     0       0       0       0
D0442   a^ r      0      0      0     0     0     0     0       0       0       0
D0443   a^ r      0      0      0     0     0     0     0       0       0       0
```

```
                                                    03-FEB-1996 16:53:26
         23 Rq/S                                         Up:    0 0:25.45
ASWE   KB/S   Rd%   Wr%   Cm%   HT%   PH%   MS%   Purge   BlChd   BlHit
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
a^ r      0     0     0     0     0     0     0       0       0       0
```

Figure 5–4 VTDPY Brief SCSI Status Display

```
VTDPY Monitor
                    60.9% Idle                 861 KB/S       293 Rq/S              Up:     0    1:33.50
  Pr     Name     Stk/Max    Typ    Sta    CPU%      Target   Unit    ASWC     KB/S   Unit    ASWC    KB/S
   0     NULL       0/  0           Rn     60.9    01234567   D0000   o^ b       98
   2    RECON      10/  1    FNC    Bl      0.0    P1D D D  H D0001   o^ b      100
   3     SHIS      40/  7    FNC    Rn     32.7    o2 D D DDH D0002   o^ b      115
   8    VTDPY      10/  3    CUP    Rn      0.1    r3D D D  H D0003   o^ b       86
  18   SCSIVT      10/  1    FNC    Bl      0.0    t4 D D  DH D0004   o^ b       96
  19    DS_HB      10/  1    FNC    Bl      0.0     5D D D  H D0005   o^ b      141
  24       VA      10/  1    FNC    Bl      0.0     6 D D DDH D0006   o^ b      125
  25     DS_1      40/  6    FNC    Bl      5.8               D0007   o^ b       97
  26     DS_0      20/  1    FNC    Bl      0.1   Xfer Rate
  27  CLIMAIN      16/  7    FNC    Bl      0.0   Id   Mhz
  28    NVFOC      10/  1    FNC    Bl      0.0   0 = Asynch
  29   REMOTE      10/  1    FNC    Bl      0.0   1 = Asynch
  30      FOC      20/  2    FNC    Bl      0.0   2 = This
  31    DUART      10/  1    FNC    Bl      0.0   3 = Asynch
                                                  4 = Asynch
                                                  5 = Asynch
                                                  6 = 3.57
                                                  7 = Asynch
```

5.2.2.1 Display Header

```
SC4600 1 S/N: CX00000002 2 SW: V26Z 3 HW:  A-02 5
VTDPY Monitor Copyright © 1994, Digital Equipment Corp.5
```

This subdisplay provides title information for the display. For 132 column displays, this subdisplay is all on one line.

[1] Controller model
[2] Controller serial number
[3] Controller firmware version
[4] Controller hardware version
[5] Copyright notice

5.2.2.2 Date and Time

```
  29-JAN-1994 13:46:34 1
        Up:    1  3:45.19 2
```

This subdisplay provides time information for the display.

[1] System date and time. This information is not displayed for SCSI based HS controllers.
[2] Time in days, hours, minutes, and seconds since the last controller boot.

5.2.2.3 Controller Performance Summary

```
  47.2% Idle 1 1225 KB/S 2106 Rq/S 3
```

This subdisplay provides total system performance information.

[1] Policy processor idle rate.
[2] Cumulative data transfer rate in kilobytes per second. When logical units are being displayed, this is the transfer rate between the host and the controller. When physical devices are being displayed, this is the transfer rate between the controller and the devices.
[3] Cumulative unit or device request rate per second.When logical units are being displayed, this is the request rate between the host and the controller. When physical devices are being isplayed, this is the request rate between the controller and the devices.

5.2.2.4 Controller Threads Display

Pr[1]	Name[2]	Stk/Max[3]	Typ[4]	Sta[5]	CPU%[6]
0	NULL	0/ 0		Rn	47.2
3	HPT	40/ 7	FNC	Rn	40.3
8	VTDPY	10/ 3	DUP	Rn	0.1
18	FMTHRD	10/ 2	FNC	Bl	0.0
19	DS_HB	10/ 2	FNC	Bl	0.0
20	DUP	10/ 2	FNC	Bl	1.3
21	SCS	10/ 2	FNC	Bl	0.0
22	MSCP	20/ 6	FNC	Bl	0.0
24	VA	10/ 3	FNC	Bl	1.2
25	DS_1	40/ 6	FNC	Rn	8.9
26	DS_0	20/ 4	FNC	Bl	0.0
27	HIS	10/ 2	FNC	Bl	0.0
28	CLIMAIN	16/ 6	FNC	Bl	0.0
30	FOC	16/ 4	FNC	Bl	0.0
31	DUART	10/ 2	FNC	Bl	0.0

This display shows the status and characteristics of the active threads in the controller. Threads that are not active, such as DUP Local Program threads are not displayed until they become active. If the number of active threads exceeds the available space, not all of them will be displayed.

1 The **Pr** column lists the thread priority. The higher the number, the higher the priority.

2 The **Name** column contains the thread name. For DUP Local Program threads, this is the name used to invoke the program.

3 The **Stk** column lists the allocated stack size in 512 byte pages. The **Max** column lists the number of stack pages actually used.

4 The **Typ** column lists the thread type. The following thread types may appear:

- **FNC—Functional thread. Those threads that are started when the controller boots and never exit.**
- **DUP—DUP local program threads. These threads are only active when run either from a DUP connection or through the command line interpreter's RUN command.**
- **NULL—The NULL thread does not have a thread type because it is a special type of thread that only executes when no other thread is executable.**

5 The **Sta** column lists the current thread state. The following thread states may appear:

- **Bl—The thread is blocked waiting for timer expiration, resources, or a synchronization event.**
- Io**—A DUP local program is blocked waiting for terminal I/O completion.**
- Rn**—The thread is currently executable.**

6 The CPU% column lists the percentage of execution time credited to each thread since th e fact that there may not be enough room to display all of the threads. An unexpected amount of time may be credited to some threads because the controller's firmware architecture allows code from one thread to execute in the context of another thread without a context switch.

Table 5–5 describes the processes that may appear in the active thread display.

NOTE

It is possible that different versions of the controller firmware will have different threads or different names for the threads.

Table 5–5 Thread Description

Thread Name	Description
CLI	A local program that provides an interface to the controller's command line interpreter thread.
CLIMAIN	The command line interpreter (CLI) thread.
CONFIG	A local program that locates and adds devices to an SC–4000-series controller configuration.
DILX	A local program that exercises disk devices.
DIRECT	A local program that returns a listing of available local programs.
DS_0	A device error recovery management thread.
DS_1	The thread that handles successful completion of physical device requests.
DS_HB	The thread that manages the device and controller error indicator lights and port reset buttons.
DUART	The console terminal interface thread.
DUP	The DUP protocol server thread.
FMTHREAD	The thread that performs error log formatting and fault reporting for the controller.
FOC	The thread that manages communication between the controllers in a dual controller configuration.
MDATA	The thread that processes metadata for nontransportable disks.
NULL	The process that is scheduled when no other process can be run.
NVFOC	The thread that initiates state change requests for the other controller in a dual controller configuration. Not applicable to the StorageWorks RAID Array 310.
REMOTE	The thread that manages state changes initiated by the other controller in a dual controller configuration. Not applicable to the StorageWorks RAID Array 310.
RMGR	The thread that manages the data buffer pool.
RECON	The thread that rebuilds the parity blocks on RAID 5 storagesets when needed and manages mirrorset copy operations when necessary.
SCS	The SCS directory thread.
SCSIVT	A thread that provides a virtual terminal connection to the CLI over the host SCSI bus.
SHIS	The host SCSI protocol interface thread for SCSI controllers.
TILX	A local program that exercises tape devices.
VA	The thread that provides host protocol independent logical unit services.
VTDPY	A local program thread that provides a dynamic display of controller configuration and performance information.

5.2.2.5 SCSI Host Port Characteristics

```
Xfer Rate
T¹ W²I³Mhz⁴
1 W 7 10.00
2 W Async⁵
```

This subdisplay shows the current host port SCSI target identification, any initiator which has negotiated synchronous transfers, and the negotiated transfer method currently in use between the controller and the initiators. This subdisplay is available only for SCSI based HS controllers.

[1] SCSI host port target ID.

[2] Transfer width. **W** indicates 16 bit or wide transfers are being used. A space indicates 8 bit transfers are being used.

[3] The initiator with which synchronous communication has been negotiated.

[4] A numeric value indicates the synchronous data rate which has been negotiated with the initiator at the specified SCSI ID. The value is listed in megahertz (Mhz). In this example, the negotiated synchronous transfer rate is approximately 3.57 Mhz. To convert this number to the nanosecond period, invert and multiply by 1000. The period for this is approximately 280 nanoseconds.

[5] **Async** indicates communication between this target and all initiators is being done in asynchronous mode. This is the default communication mode and is used unless the initiator successfully negotiates for synchronous communications. If there is no communication with a given target ID, the communication mode is listed as asynchronous .

5.2.2.6 Device SCSI Status

```
Target
  01234567 ❶
P1 DDDDFhH ❷
o2TTT T hH
r3DDD   hH
t4DDDDDDhH
 5DDDD  hH
 6      hH❸
```

This display shows what devices the controller has been able to identify on the device busses.

NOTE

The controller does not look for devices that are not configured into the nonvolatile memory using the CLI ADD command.

[1] The column headings indicate the SCSI target numbers for the devices. SCSI targets are in the range 0 through 7. Target 7 is always used by a controller. In a dual controller configuration, target 6 is used by the second controller.

[2] The device grid contains a letter signifying the device type in each port/target location where a device has been found:

- D **indicates a disk device.**
- F **indicates a device type not listed above.**
- H **indicates bus position of this controller.**
- h **indicates bus position of the other controller.**
- **A period (.) indicates the device type is unknown.**
- **A space indicates there is no device configured at this location.**

[3] This subdisplay contains a row for each SCSI device port supported by the controller. The subdisplay for a controller that has six SCSI device ports is shown.

5.2.2.7 Unit Status (ABBREVIATED)

```
Unit¹   ASWC² KB/S³ Rd%⁴  Wr%⁵  Cm%⁶  HT%⁷
D0110   a^ r   0     0     0     0     0
D0120   a^ r   0     0     0     0     0
D0130   o^ r  236   100    0     0    100
```

This subdisplay shows the status of the logical units that are known to the controller firmware. It also indicates performance information for the units. Up to 42 units can be displayed in this subdisplay.

1 The **Unit** column contains a letter indicating the type of unit followed by the unit number of the logical unit. The list is sorted by unit number. The following device types may appear.

- D **indicates a disk device**
- F **indicates a device type not listed**
- U **indicates the device type is unknown**

2 The **ASWC** colulmns indicate, respectively, the availability, spindle state, write protect state, and cache state of the logical unit.

The availability state is indicated using the following letters:

- a**—Available. Available to be mounted by a host system.**
- d**—Offline, Disabled by Digital Multivendor Customer Services. The unit has been disabled for service.**
- e**—Online, Exclusive Access. Unit has been mounted for exclusive access by a user.**
- f**—Offline, Media Format Error. The unit cannot be brought available due to a media format inconsistency.**
- i**—Offline, Inoperative. The unit is inoperative and cannot be brought available by the controller.**
- m**—Offline, Maintenance. The unit has been placed in maintenance mode for diagnostic or other purposes.**
- o**—Online. Mounted by at least one of the host systems.**
- r**—Offline, Rundown. The CLI SET NORUN command has been issued for this unit.**
- v**—Offline, No Volume Mounted. The device does not contain media.**
- x**—Online to other controller. Not available for use by this controller.**
- **A space in this column indicates the availability is unknown.**

The spindle state is indicated using the following characters:

- ^**—For disks, this symbol indicates the device is at speed. For tapes, it indicates the tape is loaded.**
- >**—For disks, this symbol indicates the device is spinning up. For tapes, it indicates the tape is loading.**
- <**—For disks, this symbol indicates the device is spinning down. For tapes, it indicates the tape is unloading.**
- v**—For disks, this symbol indicates the device is stopped. For tapes, it indicates the tape is unloaded.**
- **For other types of devices, this column is left blank.**

For disks and tapes, a **w** in the write protect column indicates the unit is write protected. This column is left blank for other device types.
The data caching state is indicated using the following letters:

- b**—Both Read caching and Write Back caching are enabled.**
- r**—Read caching is enabled.**
- w**—Write Back caching is enabled.**

- **A space in this column indicates caching is disabled.**

3 **KB/S**—This column indicates the average amount of kilobytes of data transferred to and from the unit in the previous screen update interval. This data is available only for disk and tape units.

4 **Rd%**—This column indicates what percentage of data transferred between the host and the unit were read from the unit. This data is contained only in the **DEFAULT** display for disk and tape device types.

5 **Wr%—**This column indicates what percentage of data transferred between the host and the unit were written to the unit. This data is contained only in the **DEFAULT** display for disk and tape device types.

6 **Cm%**—This column indicates what percentage of data transferred between the host and the unit were compared. A compare operation can be accompanied by either a read or a write operation, so this column is not cumulative with read percentage and write percentage columns. This data is contained only in the **DEFAULT** display for disk and tape device types.

7 **HT%**—This column indicates the cache hit percentage for data transferred between the host and the unit.

5.2.2.8 Unit Status (FULL)

Unit[1]	ASWC[2]	KB/S[3]	Rd%[4]	Wr%[5]	Cm%[6]	HT%[7]	PH%[8]	MS%[9]	Purge[10]	BlChd[11]	BlHit[12]
D0003	o^ r	382	0	100	0	0	0	0	0	6880	0
D0250	o^ r	382	100	0	0	0	0	100	0	6880	0
D0251	o^ r	284	100	0	0	0	0	100	0	5120	0\
D0262	a^ r	0	0	0	0	0	0	0	0	0	0
D0280	o^ r	497	44	55	0	0	0	100	0	9011	0
D0351	a^ r	0	0	0	0	0	0	0	0	0	0
D0911	a^ r	0	0	0	0	0	0	0	0	0	0
D1000	a^ r	0	0	0	0	0	0	0	0	0	0

This subdisplay shows the status of the logical units that are known to the controller firmware. It also shows I/O performance information and caching statistics for the units. Up to 42 units can be displayed in this subdisplay.

1 The Unit column contains a letter indicating the type of unit followed by the unit number of the logical unit. The list is sorted by unit number. There may be duplication of unit numbers between devices of different types. If this happens, the order of these devices is arbitrary. The following device type letters may appear:

- D **indicates a disk device.**
- F **indicates a device type not listed above.**
- U **indicates the device type is unknown.**

2 The **ASWC** columns indicate the availability, spindle state, write protect state, and cache state respectively of the logical unit.

The availability state is indicated using the following letters:

- a**—Available. Available to be mounted by a host system.**
- d**—Offline, Disabled by Digital Multivendor Customer Services. The unit has been disabled for service.**
- e**—Online, Exclusive Access. Unit has been mounted for exclusive access by a user.**
- f**—Offline, Media Format Error. The unit cannot be brought available due to a media format inconsistency.**

- i**—Offline, Inoperative. The unit is inoperative and cannot be brought available by the controller.**
- m**—Offline, Maintenance. The unit has been placed in maintenance mode for diagnostic or other purposes.**
- o**—Online. Mounted by at least one of the host systems.**
- r**—Offline, Rundown. The CLI SET NORUN command has been issued for this unit.**
- v**—Offline, No Volume Mounted. The device does not contain media.**
- x**—Online to other controller. Not available for use by this controller.**
- **A space in this column indicates the availability is unknown.**

The spindle state is indicated using the following characters:

- ^**—For disks, this symbol indicates the device is at speed. For tapes, it indicates the tape is loaded.**
- >**—For disks, this symbol indicates the device is spinning up. For tapes, it indicates the tape is loading.**
- <**—For disks, this symbol indicates the device is spinning down. For tapes, it indicates the tape is unloading.**
- v**—For disks, this symbol indicates the device is stopped. For tapes, it indicates the tape is unloaded.**

– For other types of devices, this column is left blank.

For disks, a **w** in the write protect column indicates the unit is write protected. This column is left blank for other device types.

The data caching state is indicated using the following letters:

- b**—Both Read caching and Write Back caching are enabled.**
- r**—Read caching is enabled.**
- w**—Write Back caching is enabled.**
- **A space in this column indicates caching is disabled.**

3 **KB/S**—This column indicates the average amount of kilobytes of data transferred to and from the unit in the previous screen update interval. This data is only available for disk and tape units.

4 **Rd%**—This column indicates what percentage of data transferred between the host and the unit were read from the unit. This data is only contained in the **DEFAULT** display for disk and tape device types.

5 **Wr%**—This column indicates what percentage of data transferred between the host and the unit were written to the unit. This data is only contained in the **DEFAULT** display for disk and tape device types.

6 **Cm%—**This column indicates what percentage of data transferred between the host and the unit were compared. A compare operation may be accompanied by either a read or a write operation, so this column is not cumulative with read percentage and write percentage columns. This data is only contained in the **DEFAULT** display for disk and tape device types.

7 **HT%**—This column indicates the cache hit percentage for data transferred between the host and the unit.

8 **PH%**—This column indicates the partial cache hit percentage for data transferred between the host and the unit.

9 **MS%**—This column indicates the cache miss percentage for data transferred between the host and the unit.

10 **Purge—**This column shows the number of blocks purged from the write back cache in the last update interval.

11 **BlChd**—This column shows the number of blocks added to the cache in the last update interval.

12 **BlHit**—This column shows the number of cached data blocks "hit" in the last update interval.

5.2.2.9 Device Status

PTL[1]	ASWF[2]	Rq/S[3]	RdKB/S[4]	WrKB/S[5]	Que[6]	Tg[7]	CR[8]	BR[9]	TR[10]
D100	A^	0	0	0	11	0	0	0	0
D120	A^	0	0	0	0	0	0	0	0
D140	A^	0	0	0	0	0	0	0	0
D210	A^	11	93	0	1	1	0	0	0
D230	A^	0	0	0	0	0	0	0	0

This subdisplay shows the status of the physical storage devices that are known to the controller firmware. It also shows I/O performance information and bus statistics for these devices. Up to 42 devices can be displayed in this subdisplay.

1 The **PTL** column contains a letter indicating the type of device followed by the SCSI Port, Target, and LUN of the device. The list is sorted by port, target, and LUN. The following device type letters may appear:

- D **indicates a disk device.**
- F **indicates a device type not listed above.**
- U **indicates the device type is unknown.**

2 The **ASWF** columns indicate the allocation, spindle state, write protect state, and fault state respectively of the device.

The availability state is indicated using the following letters:

- A**—Allocated to this controller.**
- a**—Allocated to the other controller.**
- U**—Unallocated, but owned by this controller.**
- u**—Unallocated, but owned by the other controller.**
- **A space in this column indicates the allocation is unknown.**

The spindle state is indicated using the following characters:

- ^**—For disks, this symbol indicates the device is at speed. For tapes, it indicates the tape is loaded.**
- >**—For disks, this symbol indicates the device is spinning up. For tapes, it indicates the tape is loading.**
- <**—For disks, this symbol indicates the device is spinning down. For tapes, it indicates the tape is unloading.**
- v**—For disks, this symbol indicates the device is stopped. For tapes, it indicates the tape is unloaded.**
- **For other types of devices, this column is left blank.**

For disks and tapes, a **W** in the write protect column indicates the device is hardware write protected. This column is left blank for other device types.

A **F** in the fault column indicates an unrecoverable device fault. If this field is set, the device fault indicator also is illuminated.

3 **Rq/S**—This column shows the average I/O request rate for the device during the last update interval. These requests are up to eight kilobytes long and are either generated by host requests or cache flush activity.

4 **RdKB/S**—This column shows the average data transfer rate from the device in kilobytes during the previous screen update interval.

5 **WrKB/S**—This column shows the average data transfer rate to the device in kilobytes during the previous screen update interval.

6 **Que**—This column shows the maximum number of transfer requests waiting to be transferred to the device during the last screen update interval.

7 **Tg**—This column shows the maximum number of transfer requests queued to the device during the last screen update interval. If a device does not support tagged queuing, the maximum value is 1.

[8] **CR**—This column indicates the number of SCSI command resets that occurred since VTDPY was started.

[9] **BR**—This column indicates the number of SCSI bus resets that occurred since VTDPY was started.

[10] **TR**—This column indicates the number of SCSI target resets that occurred since VTDPY was started.

5.2.2.10 Device SCSI Port Performance

```
Port[1]  Rq/S[2]  RdKB/S[3]   WrKB/S[4]    CR[5]  BR[6]  TR[7]
  1       0        0           0          0      0      0
  2      11       93           0          0      0      0
  3      48      341           0          0      0      0
  4      48      340           0          0      0      0
  5      58       93         375          0      0      0
  6       0        0           0          0      0      0
```

This subdisplay shows the accumulated I/O performance values and bus statistics for the SCSI device ports. The subdisplay for a controller that has six SCSI device ports in shown.

[1] The **Port** column indicates the number of the SCSI device port.

[2] **Rq/S**—This column shows the average I/O request rate for the port during the last update interval. These requests are up to eight kilobytes long and are either generated by host requests or cache flush activity.

[3] **RdKB/S**—This column shows the average data transfer rate from all devices on the SCSI bus in kilobytes during the previous screen update interval.

[4] **WrKB/S**—This column shows the average data transfer rate to all devices on the SCSI bus in kilobytes during the previous screen update interval.

[5] **CR**—This column indicates the number of SCSI command resets that occurred since VTDPY was started.

[6] **BR**—This column indicates the number of SCSI bus resets that occurred since VTDPY was started.

[7] **TR**—This column indicates the number of SCSI target resets that occurred since VTDPY was started.

5.2.2.11 Help Example

```
VTDPY> HELP

Available VTDPY commands:

^C - Prompt for commands
^G or ^Z - Update screen
^O - Pause/Resume screen updates
^Y - Terminate program^R or ^W - Refresh screen
DISPLAY CACHE - Use 132 column unit caching statistics display
DISPLAY DEFAULT - Use default 132 column system performancedisplay
DISPLAY DEVICE - Use 132 column device performance display
DISPLAY STATUS - Use 80 column controller status display
EXIT - Terminate program (same as QUIT)
INTERVAL <seconds> - Change update interval
HELP - Display this help message
REFRESH - Refresh the current display
QUIT - Terminate program (same as EXIT)
UPDATE - Update screen display

VTDPY>
```

This is the sample output from executing the HELP command.

5.3 Exercising the Subsystem

> **NOTE**
>
> Some examples in this chapter may include references to tape, tape loader, or CDROM devices. SC–4000-series controllers do not support such devices. Inclusion of examples referencing tape, tape loader, and CDROM devices does not imply controller support.

Use the disk inline exerciser (DILX) diagnostic tool to exercise the data transfer capabilities of selected disks installed in your subsystem. DILX exercises disks in a way that simulates a high level of user activity. You can use the DILX utility to determine the health of a controller and the disks connected to it. You can also use it to gather subsystem performance statistics.

5.3.1 Special Considerations

Be aware of the following special considerations when running DILX:

You can run DILX from a maintenance terminal or virtual host terminal.

DILX allows for autoconfiguring of drives. This allows for quick testing of all disk units simultaneously. Be aware that *customer data will be lost* by running this test. Digital recommends using the Auto-Configure option only during initial installations, before customer data has been accumulated on the subsystem.

DILX acts only on disks configured as single-device units. Error reports identify the logical units, not the physical devices. An attempt to run DILX on a RAIDset, stripeset, or mirrorset results in an error message.

There are no limitations on the number of units DILX may test at one time. However, Digital recommends only using DILX when no host activity is present. If you must run DILX during a live host connection, you should limit your testing to no more than half of any controller's units at one time. This conserves controller resources and minimizes performance degradation on the live units you are not testing.

5.3.2 Invoking DILX

To invoke the DILX utility from a maintenance terminal, type:

```
HSZ20> RUN DILX
```

5.3.3 Interrupting DILX Execution

Use the following guidelines to interrupt DILX execution.

> **NOTE**
>
> The symbol "^"is equivalent to the Ctrl key. You must press and hold the Ctrl key and type the character key given.

Ctrl/G or Ctrl/T causes DILX to produce a performance summary. DILX continues normal execution without affecting the runtime parameters.

Ctrl/C causes DILX to produce a performance summary, stop testing, and ask the "reuse parameters" question.

Ctrl/Y causes DILX to abort. The "reuse parameters" question is not asked.

5.3.4 Running DILX Tests

DILX offers the following tests:

- The Basic Function test
- The User-Defined test

5.3.4.1 Running the Basic Function Test

The Basic Function test for DILX executes in two or three phases. The three phases are as follow:

1. **Initial Write Pass**—Is the only optional phase and is always executed first (if selected). The initial write pass writes the selected data patterns to the entire specified data space or until the DILX execution time limit has been reached. Once the initial write pass has completed, it is not reexecuted no matter how long the DILX execution time is set. The other phases are reexecuted on a 10-minute cycle.
2. **Random I/O**—Simulates typical I/O activity with random transfers from one byte to the maximum size I/O possible with the memory constraints DILX runs under. Note that the length of all I/Os is in bytes and is evenly divisible by the sector size (512 bytes).

   Read and write (if enabled) commands are issued using random logical block numbers (LBNs). In the read/write mode, DILX issues the read and write commands in the ratio specified previously under read/write ratio. When read-only mode is chosen, only read commands are issued.

   If compare operations are enabled, they are performed on read commands using DILX internal checks. The percentage of compares to perform can be specified. This phase is executed 80 percent of the time. It is the first phase executed after the initial write pass has completed. It is reexecuted at 10-minute intervals with each cycle lasting approximately 8 minutes.

   Intervals are broken down into different cycles. The interval is repeated until the user-selected time interval expires.

```
<---------------------------10 min----------------------------->
  <------------8 min Random I/O----------><--2 min Data Inten--->
```

3. **Data Intensive**—Designed to test disk throughput by selecting a starting LBN and repeating transfers to the next sequential LBN that has not been accessed by the previous I/O. The transfer size of each I/O equals the maximum sized I/O that is possible with the memory constraints DILX must run under. This phase continues performing spiraling I/O to sequential tracks. Read and write commands are issued in read/write mode. This phase is executed 20 percent of the time after the initial write pass has completed. This phase always executes after the random I/O phase. It is reexecuted at 10-minute intervals with each cycle approximately 2 minutes.

5.3.4.2 User-Defined Test

CAUTION

The User-Defined test must be run *only* by very knowledgeable personnel. Otherwise, customer data can be destroyed.

When this test is selected, DILX prompts you for input to define a specific test. In the DILX User-Defined test, a total of 20 or fewer I/O commands can be defined. Once all of the commands are issued, DILX issues the commands again in the same sequence. This is repeated until the selected time limit is reached. As you build the test, DILX collects the following information from you for each command:

- The I/O command name (write, read, or quit). Quit is not a command; instead it indicates to DILX that you have finished defining the test.
- The starting logical block number (LBN).
- The size of the I/O in 512 byte blocks.

5.3.5 Understanding Test Definition Questions

The following text is displayed when running DILX. The text includes questions that are listed in the approximate order that they are displayed on your terminal. These questions prompt you to define the runtime parameters for DILX.

NOTE

Defaults for each question are given inside []. If you press the Return key as a response to a question, the default is used as the response.

After DILX has been started, the following message and prompt is displayed:

Message:

```
It is recommended that DILX only be run when there is no host
activity present on the SC-4000-series controller. Do you want to
continue (y/n) [n] ?
```

Explanation: The program allows you to exit if you haven't halted host activity on the controller.

The following messages describing the Auto-Configure option are displayed:

Message:

```
The Auto-Configure option will automatically select, for testing,
half or all of the disk units configured. It will perform a very
thorough test with *WRITES* enabled. Only disk units with a single
physical device will be tested. The user will only be able to
select the run time and performance summary options and whether to
test a half or full configuration. The user will not be able to
specify specific units to test. The Auto-Configure option is only
recommended for initial installations.

Do you wish to perform an Auto-Configure (y/n) [n] ?
```

Explanation: Enter "Y" if you wish to invoke the Auto-Configure option.

After the Auto-Configure option is selected, DILX will display the following caution statement:

Message:

```
 **CAUTION**
All data on the Auto-Configured disks will be destroyed. You
*MUST* be sure of yourself.

Are you sure you want to continue (y/n) [n] ?
```

Explanation: This question is self explanatory.

Message:

```
Use All Defaults and Run in Read Only Mode (y/n)[y]?
```

Explanation: Enter "Y" to use the defaults for DILX, run in read-only mode, and most of the other DILX questions are not asked. Enter "N" and the defaults are not used. You must then answer each question as it is displayed. The following defaults are assumed for all units selected for testing:

- Execution time limit is 10 minutes.
- Performance summary interval is 10 minutes.
- Displaying sense data for hard or soft errors is disabled.
- The hard error limit is 65535. Testing will stop if the limit is reached.
- The I/O queue depth is 4. A maximum of 4 I/Os will be outstanding at any time.
- The selected test is identical to the Basic Function test.
- Read-only mode.
- All user available LBNs are available for testing.
- Data compares are disabled.

Message:

```
Enter the execution time limit in minutes (1:65535)[10]?
```

Explanation: Enter the desired time you want DILX to run. The default run time is 10 minutes.

Message:

```
Enter performance summary interval in minutes (1:65535)[10]?
```

Explanation: Enter a value to set the interval for which a performance summary is displayed. The default is 10 minutes.

Message:

```
Include performance statistics in performance summary (y/n)[n]?
```

Explanation: Enter "Y" to see a performance summary that includes the performance statistics that include the total count of read and write I/O requests and the kilobytes transferred for each command type. Enter "N" and no performance statistics are displayed.

Message:

```
Display hard/soft errors (y/n)[n]?
```

Explanation: Enter "Y" to enable displays of sense data and deferred errors. Enter "N" to disable error reporting. The default is disabled error reporting.

Message:

```
When the hard error limit is reached, the unit will be dropped
from testing.
Enter hard error limit (1:65535) [65535] ?
```

Explanation: Enter a value to specify the hard error limit for all units to test. This question is used to obtain the hard error limit for *all* units under test. If the hard error limit is reached, DILX discontinues testing the unit that reaches the hard error limit. If other units are currently being tested by DILX, testing continues for those units.

Message:

```
When the soft error limit is reached, soft errors will no longer
be displayed but testing will continue for the unit. Enter soft
error limit (1:65535) [32] ?
```

Explanation: Enter a value to specify the soft error limit for *all* units under test. When the soft error limit is reached, soft errors are no longer displayed, but testing continues for the unit.

Message:

```
Enter IO queue depth (1:12) [4]?
```

Explanation: Enter the maximum number of outstanding I/Os for each unit selected for testing. The default is 4.

Message:

```
Enter unit number to be tested?
```

Explanation: Enter the unit number for the unit to be tested.

NOTE

When DILX asks for the unit number, it requires the number designator for the disk, where, for example, D117 would be specified as unit number 117.

Message:

```
Unit x will be write enabled.
 Do you still wish to add this unit (y/n) [n]?
```

Explanation: This is a reminder of the consequences of testing a unit while it is write enabled. This is the last chance to back out of testing the displayed unit. Enter "Y" to write enable the unit. Enter "N" to back out of testing that unit.

Message:

```
Select another unit (y/n) [n]?
```

Explanation: Enter “Y” to select another unit for testing. Enter “N” to begin testing the units already selected. The system will display the following test selections:

Message:

```
***Available tests are:
1. Basic Function
2. User Defined Test
Use the Basic Function 99.9% of the time. The User Defined
test is for special problems only.
Enter test number (1:2) [1]?
```

Explanation: Enter “1” for the Basic Function test or “2” for the User-Defined test. After selecting a test, the system will then display the following messages:

Message:

```
In the User-Defined test, you may define up to 20 commands. They
will be executed in the order entered. The commands will be
repeated until the execution time limit expires.

** CAUTION **
If you define write commands, user data will be destroyed.

Enter command number x (read, write, quit) []?
```

Explanation: This question only applies to the User-Defined test. It allows you to define command *x* as a read or write command. Enter quit to finish defining the test.

After making your command selections, the following message is displayed by DILX:

Message:

```
* IMPORTANT * If you answer yes to the next question, user data
WILL BE destroyed.

Write enable disk unit (y/n) [n] ?
```

Explanation: Enter “Y” to write enable the unit. Write commands are enabled for the currently selected test. Data within your selected LBN range will be destroyed. *Be sure of your actions before answering this question.* This question applies to all DILX tests. Enter “N” to enable read only mode, in which read and access commands are the only commands enabled.

Message:

```
Perform initial write (y/n) [n] ?
```

Explanation: Enter "Y" to write to the entire user-selected LBN range with the user-selected data patterns. Enter "N" for no initial write pass.

If you respond with "Y," the system performs write operations starting at the lowest user-selected LBN and issues spiral I/Os with the largest byte count possible. This continues until the specified LBN range has been completely written. Upon completion of the initial write pass, normal functions of the Random I/O phase start. The advantage of selecting the initial write pass is that compare host data commands can then be issued and the data previously written to the media can be verified for accuracy. It makes sure that all LBNs within the selected range are accessed by DILX.

The disadvantage of using the initial write pass is that it may take a long time to complete, because a large LBN range was specified. You can bypass this by selecting a smaller LBN range, but this creates another disadvantage in that the entire disk space is not tested. The initial write pass only applies to the Basic Function test.

Message:

```
The write percentage will be set automatically.\ Enter read
percentage for random IO and data intensive phase (0:100) [67] ?
```

Explanation: This question is displayed if read/write mode is selected. It allows you to select the read/write ratio to use in the Random I/O and Data Intensive phases. The default read/write ratio is similar to the I/O ratio generated by a typical OpenVMS system.

Message:

```
Enter data pattern number 0=all, 19=user_defined, (0:19) [0] ?
```

Explanation: The DILX data patterns are used in write commands. This question is displayed when write operations are enabled for the Basic Function or User-Defined tests. There are 18 unique data patterns to select from. These patterns were carefully selected as worst case or most likely to produce errors for disks connected to the controller. (See Section 5.3.9 for a list of data patterns.) The default uses all 18 patterns in a random method. This question also allows you to create a unique data pattern of your own choice.

Message:

```
Enter the 8-digit hexadecimal user defined data pattern [] ?
```

Explanation: This question is only displayed if you choose to use a User-Defined data pattern for write commands. The data pattern is represented in a longword and can be specified with eight hexadecimal digits.

Message:

```
Enter start block number (0:highest_lbn_on_the_disk) [0] ?
```

Explanation: Enter the starting block number of the area on the disk you wish DILX to test. Zero is the default.

Message:

```
Enter end block number (starting_lbn:highest_lbn_on_the_disk)
[highest_lbn_on_the_disk] ?
```

Explanation: Enter the highest block number of the area on the disk you wish DILX to test. The highest block number (of that type of disk) is the default.

Message:

```
Perform data compare (y/n) [n] ?
```

Explanation: Enter “Y” to enable data compares. Enter “N” and no data compare operations are done.

This question is only asked if you select the initial write option. Data compares are only performed on read operations. This option can be used to test data integrity.

Message:

```
Enter compare percentage (1:100) [5] ?
```

Explanation: This question is displayed only if you choose to perform data compares. This question allows you to change the percentage of read and write commands that will have a data compare operation performed. Enter a value indicating the compare percentage. The default is 5.

Message:

```
Enter command number x (read, write, quit) [] ?
```

Explanation: This question only applies to the User-Defined test. It allows you to define command *x* as a read, write, access, or erase command. Enter quit to finish defining the test.

Message:

```
Enter starting LBN for this command (0:highest_lbn_on_the_disk) []
?
```

Explanation: This question only applies to the User-Defined test. It allows you to set the starting LBN for the command currently being defined. Enter the starting LBN for this command.

Message:

```
Enter the IO size in 512 byte blocks for this command
(1:size_in_blocks) [] ?
```

Explanation: This question only applies to the User-Defined test. It allows you to set the I/O size in 512-byte blocks for the command currently being defined. Enter values indicating the I/O size for this command.

Message:

```
Reuse parameters (stop, continue, restart, change_unit) [stop] ?
```

Explanation: This question is displayed after the DILX execution time limit expires, after the hard error limit is reached for every unit under test, or after you enter Ctrl/C. These options are as follow:

- **Stop**—DILX terminates normally.
- **Continue**—DILX resumes execution without resetting the remaining DILX execution time or any performance statistics. If the DILX execution time limit has expired, or all units have reached their hard error limit, DILX terminates.
- **Restart**—DILX resets all performance statistics and restarts execution so that the test will perform exactly as the one that just completed. However, there is one exception. If the previous test was the Basic Function test with the initial write pass and the initial write pass completed, the initial write pass is not performed when the test is restarted.
- **Change_unit**—DILX allows you to drop or add units to testing. For each unit dropped, another unit must be added until all units in the configuration have been tested. The unit chosen will be tested with the same parameters that were used for the unit that was dropped from testing. When you have completed dropping and adding units, all performance statistics are initialized and DILX execution resumes with the same parameters as the last run.

Message:

```
Drop unit #x (y/n) [n] ?
```

Explanation: This question is displayed if you choose to change a unit as an answer to the "reuse parameters" (previous) question. Enter the unit number that you wish to drop from testing.

Message:

```
The new unit will be write enabled. Do you wish to continue (y/n)
[n] ?
```

Explanation: This question is displayed if you choose to change a unit as an answer to the "reuse parameters" question. It is only asked if the unit being dropped was write enabled. This question gives you the chance to terminate DILX testing if you do not want data destroyed on the new unit. Enter "N" to terminate DILX.

5.3.6 Understanding Output Messages

The following message is displayed when DILX is started:

Message:

```
Disk Inline Exerciser - Version 2.5
```

Explanation: This message identifies the internal program as DILX and gives the DILX software version number.

Message:

```
Change Unit is not a legal option if Auto-Configure was chosen.
```

Explanation: This message is displayed if the user selects the Auto-Configure option and selects the "change unit response" to the "reuse parameters" question. You cannot drop a unit and add a unit if all units were selected for testing.

Message:

```
DILX - Normal Termination.
```

Explanation: This message is displayed when DILX terminates under normal conditions.

Message:

```
Insufficient resources.
```

Explanation: Following this line is a second line that gives more information about the problem, which could be one of the following messages:

Message:

```
Unable to allocate memory.
```

Explanation: DILX was unable to allocate the memory it needed to perform DILX tests. You should run DILX again but choose a lower queue depth and/or choose fewer units to test.

Message:

```
Cannot perform tests.
```

Explanation: DILX was unable to allocate all of the resources needed to perform DILX tests. You should run DILX again but choose a lower queue depth and/or choose fewer units to test.

Message:

```
Unable to change operation mode to maintenance.
```

Explanation: DILX tried to change the operation mode from normal to maintenance using the SYSAP$CHANGE_STATE()routine but was not successful due to insufficient resources. This problem should not occur. If it does occur, reset the controller.

Message:

```
Unit x is not a valid disk unit for DILX testing.
```

Explanation: An attempt was made to allocate a unit for testing that does not exist on the controller.

Message:

```
Unit x successfully allocated for testing.
```

Explanation: All processes that DILX performs to allocate a unit for testing, have been completed. The unit is ready for DILX testing.

Message:

```
Unable to allocate unit.
```

Explanation: This message should be preceded by a reason why the unit could not be allocated for DILX testing.

Message:

```
DILX detected error, code x.
```

Explanation: The “normal” way DILX recognizes an error on a unit is through the reception of SCSI sense data. This loosely corresponds to an MSCP error log. However, the following are some errors that DILX will detect using internal checks without SCSI sense data:

Message:

```
Illegal Data Pattern Number found in data pattern header. Unit x
```

This is code 1. DILX read data from the disk and found that the data was not in a pattern that DILX previously wrote to the disk.

```
No write buffers correspond to data pattern Unit x.
```

This is code 2. DILX read a legal data pattern from the disk at a place where DILX wrote to the disk, but DILX does not have any write buffers that correspond to the data pattern. Thus, the data has been corrupted.

```
Read data do not match what DILX thought was written to the media.
Unit x.
```

This is code 3. DILX writes data to the disk and then reads it and compares it against what was written to the disk. This indicates a compare failure. More information is displayed to indicate where in the data buffer the compare operation failed and what the data was and should have been.

```
DILX terminated. A termination, a print summary or a reuse
parameters\ request was received but DILX is currently not testing
any units.
```

Explanation: You entered Ctrl/Y (termination request), Ctrl/G (print summary request) or Ctrl/C (reuse parameters request) before DILX had started to test units. DILX cannot satisfy the second two requests so DILX treats all of these requests as a termination request.

Message:

```
DILX will not change the state of a unit if it is not NORMAL.
```

Explanation: DILX cannot allocate the unit for testing, because it is already in Maintenance mode. (Maintenance mode can only be invoked by the firmware. If another DILX session is in use, the unit is considered in Maintenance mode.)

Message:

```
Unable to bring unit online.
```

Explanation: This message is self explanatory.

Message:

`Soft error reporting disabled. Unit` *x*.

Explanation: This message indicates that the soft error limit has been reached and therefore no more soft errors will be displayed for this unit.

Message:

`Hard error limit reached, unit` *x* `dropped from testing.`

Explanation: This message indicates that the hard error limit has been reached and the unit is dropped from testing.

Message:

```
Soft error reporting disabled for controller errors.
```

Explanation: This message indicates that the soft error limit has been reached for controller errors. Thus, controller soft error reporting is disabled.

Message:

```
Hard error limit reached for controller errors. All units dropped
from\testing.
```

Explanation: This message is self explanatory.

Message:

```
Unit is already allocated for testing.
```

Explanation: This message is self explanatory.

Message:

```
No drives selected.
```

Explanation: DILX parameter collection was exited without choosing any units to test.

Message:

```
Maximum number of units are now configured.
```

Explanation: This message is self explanatory. (Testing will start after this message is displayed.)

Message:

```
Unit is write protected.
```

Explanation: The user wants to test a unit with write and/or erase commands enabled but the unit is write protected.

Message:

```
The unit status and/or the unit device type has changed
unexpectedly.\Unit x dropped from testing.
```

Explanation: The unit status may change if the unit experienced hard errors or if the unit is disconnected. Either way, DILX cannot continue testing the unit.

Message:

```
Last Failure Information follows. This error was NOT produced by
running\ DILX. It represents the reason why the controller crashed
on the previous controller run.
```

Explanation: This message may be displayed while allocating a unit for testing. It does not indicate any reason why the unit is or is not successfully allocated, but rather represents the reason why the controller went down in the previous run. The information that follows this message is the contents of an EIP.

Message:

```
Disk unit numbers on this controller include:
```

Explanation: After this message is displayed, a list of disk unit numbers on the controller is displayed.

Message:

```
IO to unit x has timed out. DILX aborting.
```

Explanation: One of the DILX I/Os to this unit did not complete within the command timeout interval and, when examined, was found not progressing. This indicates a failing controller.

Message:

```
DILX terminated prematurely by user request.
```

Explanation: Ctrl/Y was entered. DILX interprets this as a request to terminate. This message is displayed and DILX terminates.

Message:

```
Unit is owned by another sysap.
```

Explanation: DILX could not allocate the unit specified, because the unit is currently allocated by another system application. Terminate the other system application or reset the controller.

Message:

```
This unit is reserved.
```

Explanation: The unit could not be allocated for testing, because a host has reserved the unit.

Message:

```
This unit is marked inoperative.
```

Explanation: The unit could not be allocated for testing, because the controller internal tables have the unit marked as inoperative.

Message:

```
The unit does not have any media present.
```

Explanation: The unit could not be allocated for testing, because no media is present.

Message:

```
The RUNSTOP_SWITCH is set to RUN_DISABLED.
```

Explanation: The unit could not be allocated for testing, because the RUN/NORUN switch for this unit is set to NORUN. The RUN/NORUN switch is set using the SET UNIT CLI command.

Message:

```
Unable to continue, run time expired.
```

Explanation: A continue response was given to the “reuse parameters” question. This is not a valid response if the run time has expired. Reinvoke DILX.

When DILX starts to exercise the disk units, the following message is displayed with the current time of day:

Message:

```
DILX testing started at: xx:xx:xx
    Test will run for x minutes
    Type ^T(if running DILX through a VCS) or ^G(in all other
cases)
      to get a current performance summary
    Type ^C to terminate the DILX test prematurely
    Type ^Y to terminate DILX prematurely
```

Explanation: The program presents introductory information

5.3.7 Understanding Sense Data Display

To interpret the sense data fields correctly, refer to SCSI–2 specifications. The following demonstrates a DILX sense data display:

```
Sense data in hex for unit x

      Sense Key                     x
      Sense ASC                     x
      Sense ASQ                     x
      Instance                      x
```

5.3.8 Deferred Error Display

The following demonstrates a DILX deferred error display:

```
Deferred error detected, hard error counted against each unit.
      Sense Key                     x
      Sense ASC                     x
      Sense ASQ                     x
      Instance                      x
```

5.3.9 Data Patterns

Table 5–6 defines the data patterns used with the DILX Basic Function or User-Defined tests. There are 18 unique data patterns. These data patterns were selected as worst case, or the ones most likely to produce errors on disks connected to the controller.

Table 5–6 DILX Data Patterns

Pattern Number	Pattern in Hexadecimal Numbers
1	0000
2	8B8B
3	3333
4	3091
5, shifting 1s	0001, 0003, 0007, 000F, 001F, 003F, 007F, 00FF, 01FF, 03FF, 07FF, 0FFF, 1FFF, 3FFF, 7FFF
6, shifting 0s	FIE, FFFC, FFFC, FFFC, FFE0, FFE0, FFE0, FFE0, FE00, FC00, F800, F000, F000, C000, 8000, 0000
7, alternating 1s, 0s	0000, 0000, 0000, FFFF, FFFF, FFFF, 0000, 0000, FFFF, FFFF, 0000, FFFF, 0000, FFFF, 0000, FFFF
8	B6D9
9	5555, 5555, 5555, AAAA, AAAA, AAAA, 5555, 5555, AAAA, AAAA, 5555, AAAA, 5555, AAAA, 5555, AAAA, 5555
10	DB6C
11	2D2D, 2D2D, 2D2D, D2D2, D2D2, D2D2, 2D2D, 2D2D, D2D2, D2D2, 2D2D, D2D2, 2D2D, D2D2, 2D2D, D2D2
12	6DB6
13, ripple 1	0001, 0002, 0004, 0008, 0010, 0020, 0040, 0080, 0100, 0200, 0400, 0800, 1000, 2000, 4000, 8000
14, ripple 0	FIE, FFFD, FFFB, FFF7, FFEF, FFDF, FFBF, FF7F, FEFF, FDFF, FBFF, F7FF, EFFF, BFFF, DFFF, 7FFF
15	DB6D, B6DB, 6DB6, DB6D, B6DB, 6DB6, DB6D, B6DB, 6DB6, DB6D, B6DB, 6DB6, DB6D

Pattern Number	Pattern in Hexadecimal Numbers
16	3333, 3333, 3333, 1999, 9999, 9999, B6D9, B6D9, B6D9, B6D9, FFFF, FFFF, 0000, 0000, DB6C, DB6C
17	9999, 1999, 699C, E99C, 9921, 9921, 1921, 699C, 699C, 0747, 0747, 0747, 699C, E99C, 9999, 9999
18	FFFF
Default—Use all of the above patterns in a random method	

5.3.10 Interpreting the Performance Summaries

A DILX performance display is produced under the following conditions:

- When a specified performance summary interval elapses
- When DILX terminates for any conditions except an abort
- When Ctrl/G or Ctrl/T is entered

The performance display has different formats depending on whether performance statistics are requested in the user-specified parameters and if errors are detected.

Following is an example of a DILX performance display when performance statistics were not selected and when no errors were detected:

```
DILX Summary at 18-JUN-1993 06:18:41
   Test minutes remaining: 0, expired: 6

Unit 1     Total IO Requests 482
  No errors detected
Unit 2     Total IO Requests 490
  No errors detected
```

Following is an example of a DILX performance display when performance statistics were selected and when no errors were detected:

```
   DILX Summary at 18-JUN-1993 06:18:41
   Test minutes remaining: 0, expired: 6

Unit 1     Total IO Requests 482
  Read Count   292   Write Count 168
  KB xfer   Read 7223   Write 4981   Total 12204
  No errors detected
```

Following is an example of a DILX performance display when performance statistics were not selected and when errors were detected on a unit under test:

```
   DILX Summary at 18-JUN-1993 06:18:41
   Test minutes remaining: 0, expired: 6
1Unit 10     Total IO Requests 153259
         No errors detected
2Unit 40     Total IO Requests 2161368
         Err in Hex: IC:031A4002  PTL:04/00/00  Key:04
ASC/Q:B0/00  HC:0  SC:1
         Total Errs   Hard Cnt 0   Soft Cnt 1
3Unit 55     Total IO Requests 2017193
         Err in Hex: IC:03094002  PTL:05/05/00  Key:01
ASC/Q:18/89  HC:0  SC:1
         Err in Hex: IC:03094002  PTL:05/05/00  Key:01
ASC/Q:18/86  HC:0  SC:1
4        Total Errs   Hard Cnt 0   Soft Cnt 2
```

where:

[1] Represents the unit number and the total I/O requests to this unit.

[2] Represents the unit number and total I/O requests to this unit.

All values for the following codes are described in Appendix A. This also includes the following items associated with this error, and the total number of hard and soft errors for this unit:

- **The Instance code (in hex)**
- **The port/target/LUN (PTL)**
- **The SCSI Sense Key**
- **The SCSI ASC and ASQ (ASC/Q) codes**
- **The total hard and soft count for this error**

[3] Represents information about the first two unique errors for this unit.

All values for the following codes are described in A. This also includes the following items associated with this error, and the total number of hard and soft errors for this unit:

- **The Instance code (in hex)**
- **The port/target/LUN (PTL)**
- **The SCSI Sense (Key)**
- **The SCSI ASC and ASQ (ASC/Q) codes**
- **The total hard and soft count for this error**

A line of this format may be displayed up to three times in a performance summary. There would be a line for each unique error reported to DILX for up to three errors for each unit.

[4] Represents the total hard and soft errors experienced for this unit. Following is an example of a DILX performance display when performance statistics were not selected and when a controller error was detected:

```
   DILX Summary at 18-JUN-1993 06:18:41
   Test minutes remaining: 0, expired: 6
Cnt err in HEX  IC:07080064  Key:06  ASC/Q:A0/05  HC:1  SC:0
  Total Cntrl Errs   Hard Cnt 1   Soft Cnt 0

Unit 1     Total IO Requests 482
  No errors detected
Unit 2     Total IO Requests 490
  No errors detected
```

For the previous examples, the following definitions apply. These codes are translated in A.

- IC—The Instance code.
- ASC/Q—The SCSI ASC and ASCQ code associated with this error.
- HC—The hard count of this error.
- SC—The soft count of this error.
- PTL—The location of the unit (port/target/LUN).

The performance displays contain error information for up to three unique errors. Hard errors always have precedence over soft errors. A soft error represented in one display may be replaced with information on a hard error in subsequent performance displays.

5.3.11 Abort Codes

Table 5–7 lists the DILX abort codes and definitions.

Table 5–7 DILX Abort Codes and Definitions

Value	Definition
1	An I/O has timed out.
2	dcb_p->htb_used_count reflects an available HTB to test IOs but none could be found.
3	FAO returned either FAO_BAD_FORMAT or FAO_OVERFLOW.
4	TS$SEND_TERMINAL_DATA returned either an ABORTED or INVALID_BYTE_COUNT.
5	TS$READ_TERMINAL_DATA returned either an ABORTED or INVALID_BYTE_COUNT.
6	A timer is in an unexpected expired state that prevents it from being started.
7	The semaphore was set after a oneshot I/O was issued but nothing was found in the received HTB que.
8	A termination, a print summary, or a reuse parameters request was received when DILX was not testing any units.
9	User requested an abort via Ctrl/Y.

5.3.12 Error Codes

Table 5–8 describes the DILX error codes and definitions for DILX-detected errors.

Table 5–8 DILX Error Codes and Definitions

Value	Definition
1	Illegal Data Pattern Number found in data pattern header.
2	No write buffers correspond to data pattern.
3	Read data do not match write buffer.

6

Maintaining Devices in RAID Array 310 Subsystems

This chapter describes firmware functionality related to maintaining devices in RAID Array 310 Subsystems. The RAID Array 310 firmware allows you to locate devices in the subsystem, format a SCSI device, download a device's firmware, and change volume serial numbers.

6.1 Locating Devices Displayed on the CLI in the Subsystem Cabinet

Use the CLI locate command to determine which physical device in the subsystem cabinet corresponds to a device displayed by the CLI. When you issue the locate command, the controller lights the amber fault LED on the corresponding device. You can locate a single device or all the member devices of a storageset.

To locate the physical device that corresponds to a device displayed on the CLI, follow these steps:

1. Type: `HSZ20>`**`locate 1 0 0`** (include a space between each number)
   where: 1 = the device's port address on the controller
   0 = the device's target address on the port
   0 = the device's LUN number on the target

   The controller flashes the amber faultlight of the device at the specified port, target, LUN (ptl) location.

2. Type: `HSZ20>`**`locate cancel`**
   The controller stops flashing the amber fault light on the device.

To locate all the physical devices that are member devices of a storageset, follow these steps:

1. Type: `HSZ20>`**`locate R1`**
   where r1 = the name of a storageset
   The controller flashes the amber fault lights of the member devices of the specified storageset.

2. Type: `HSZ20>`**`locate cancel`**
   The controller stops flashing the amber fault light of the member devices of the specified storageset.

6.2 Formatting a SCSI Device

Use the HSUTIL utility to perform a SCSI format operation on a disk. Once started, the HSUTIL utility performs the operation without intervention. The FORMAT_DISK option can format simultaneously up to seven disk drives.

6.2.1 Device Format Special Considerations

Be aware of the following special considerations when using the HSUTIL utility to format disks:

- HSUTIL cannot format disk drives that have been configured as single disk drive units or as members of a storageset, spareset, or failedset. If you want to format a disk drive that was previously configured as such, you will have to delete the unit number and storageset name associated with it.
- If a power failure or SCSI bus reset occurs during a format operation, the device may become unusable until a successful format operation is completed. To minimize the possibility of this problem, Digital recommends that you prevent activity to all devices on the same port as the device being formatted.
- HSUTIL cannot control or affect the defect management for a disk drive. The drive's microcode controls the defect management during formatting.
- Do not invoke any CLI command or run any local program that might reference the target disk drive(s) while HSUTIL is active.

6.2.2 Formatting a Disk Using HSUTIL

The following example demonstrates invoking the HSUTIL utility and using it to format a disk. The example formats target disk DISK110.

1. Invoke the HSUTIL utility and select the format operation by entering "1" at the prompt:

```
HSZ20> run hsutil
  *** Available functions are:
    1. FORMAT
    2. CODE_LOAD
    3. EXIT

Enter the number of the function you wish to perform (1:3) [3] ?1
```

2. The program reports the unattached units available for formatting:

```
Unattached devices on this controller include:
 Device    SCSI Product ID    Current Device Rev
 DISK110   RZ26L    (C) DEC   0001
 DISK210   ST15230N           0005
```

3. Enter the device to format ? ***disk110***
   Format DISK110 may take up to 35 minutes to format
   Select another device (y/n) [n] **Y**

   Enter the device to format ? ***disk210***
   Format DISK210 may take up to 15 minutes to format
   Select another device (y/n) [n] **N**

4. The program displays a number of informational messages. Pay particular attention to the warnings:

```
^Y and ^C will be disabled while the format operation is in progress.

CAUTION:
      When you format a device, it will destroy the data on the
      device. A backup of the device should have been done if the
      data is important.

NOTE:
      In order to minimize the possibility of a SCSI bus reset, it    is
recommended that you prevent IO operations to all other  devices on the
same port as the destination device. If a   SCSI bus reset occurs, the
format may be incomplete and you     may have to re-invoke HSUTIL.

After you answer the next question, the format will start
```

5. The program offers you the option of exiting before the format operation begins:

```
Do you want to continue(y/n)[n] ?y
```

6. The program begins the format operation, finishing a number of minutes later:

```
HSUTIL Started at: 13-JAN-1946 04:49:48
Format of DISK110 finished at 13-JAN-1946 05:45:22
Format of DISK210 finished at 13-JAN-1946 06:20:35
HSUTIL - Normal Termination at: 13-JAN-1946 06:21:43
```

7. DISK110 and DISK210 are now formatted and usable.

6.3 Updating a Device’s Firmware

Use the HSUTIL update a device’s firmware. Once started, the HSUTIL utility performs the operation without intervention

CAUTION

Device compatibility with firmware code loading varies greatly. Read the release notes for your device firmware release very carefully to make sure that the firmware is compatible with your device and HSUTIL’s code load function. Failure to ensure compatibility before using HSUTIL may result in rendering the device unusable until it is factory-initialized.

The device code load function updates a target disk’s firmware with firmware located in contiguous blocks at a specific LBN on a source disk on the same controller. The source and target disks must be configured on the controller from which HSUTIL is invoked. The source disk must be a raw disk device, configured as a unit, with no file system or label on it.

The device code load process takes place in the following two steps:

1. Preparing the Source Disk

   HSUTIL provides the capability to copy the device firmware image from a source disk *in the storage subsystem* to the target disk. The firmware image cannot be loaded directly from a location external to the controller to the target disk.

The program requires the device firmware image to be located in contiguous blocks at a known logical block number (LBN) on the source disk. You must prepare for the code load operation by copying the device firmware image to the source disk at a known LBN location.

The most common way to prepare the firmware image for downloading is to copy it to a raw source disk containing no data, as illustrated in Figure 6−1. The source disk must be a raw disk with no label and no formatted file structure on it. This process locates the firmware image in contiguous blocks at a specific LBN (usually LBN 0) on the source disk.

The details of copying the firmware image to the source disk from the host disk are specific to the host operating system.

2. Code Loading the Target Disk

   Once the firmware image is located in contiguous blocks at a known LBN on the source disk, HSUTIL is used to download it into the target disk.

6.3.1 Device Code Load Special Considerations

Be aware of the following special considerations when using the HSUTIL utility to download firmware to a disk device:

CAUTION

HSUTIL has been tested to perform the code load operation properly only with supported devices. Use HSUTIL to code load unsupported devices at your own risk.

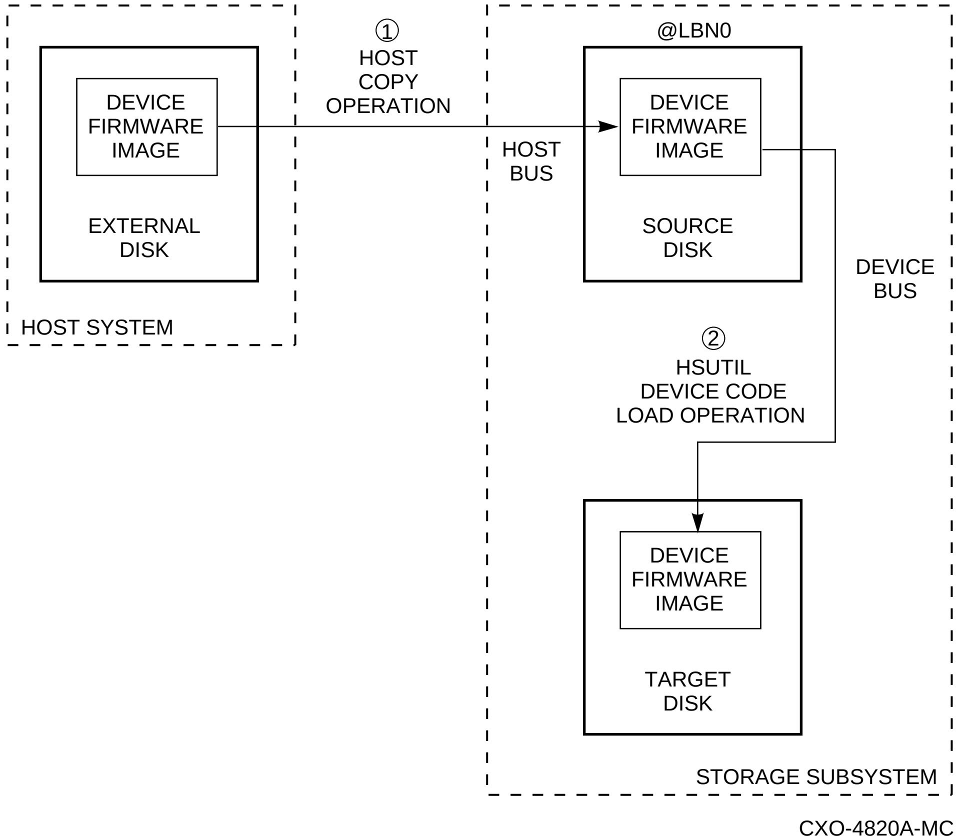
- Device compatibility with firmware code loading using the SCSI Write Buffer command varies greatly from vendor to vendor. While HSUTIL does check to see if a target disk is an unsupported device, it does not prevent you from continuing the code load operation on the device. You should understand that an unsupported device may be rendered unusable until it is factory-initialized, if the code load is unsuccessful.

CAUTION

To minimize the possibility of rendering the target device unusable, prevent bus activity to all other devices on the same port before using the HSUTIL utility to download firmware to the target device.

- If a power failure or SCSI bus reset occurs during a code load operation, permanent damage to the device can result. To minimize the possibility of rendering a device unusable, Digital recommends that you halt the activity on other devices on the same port during the code load process.

Figure 6–1 The Device Code Load Process



- HSUTIL processes a single target device per session.
- HSUTIL does not code load devices configured as units, stripesets, mirrorsets, RAIDsets, spareset, or failedset. The destination device must be an unattached device.
- The source device must be configured as a single-device unit.
- The device firmware image must be located on the source disk in contiguous blocks at a known LBN location.
- Once HSUTIL has allocated the source disk for the code load operation, the source disk is not available to other subsystem operations.
- Some device firmware releases require a format operation after a code load operation to make the device usable. In this case, you must rerun HSUTIL to perform the format operation. See the release notes for your device firmware version to determine if you must format the device after code loading. If you do not have release notes, contact the device vendor to verify whether or not a format is required after a firmware update.
- You can use the CTRL-C or CTRL-Y keys to exit HSUTIL before the program begins issuing SCSI commands to perform a code load. Once the program begins issuing SCSI commands, however, the exit key combinations are disabled.

6.3.2 Loading Code on a Supported Device (No format Required)

The following example demonstrates the use of the HSUTIL utility to download device firmware. In this case, the device's firmware release documentation specifies that it does not require a subsequent format. The example changes firmware in device DISK130 from revision 0001 to 0002. The source disk is DISK210.

The details of copying the device firmware image from a location external to the controller to the source disk are specific to the host operating system. In this example, the firmware was previously copied from the host to LBN 0 of DISK110.

1. Use the SHOW DEVICE command to report the available devices and the firmware level of each. Note that the firmware level of DISK110 is reported as "0001" and that it is an unattached device:

```
HSZ20> sho dev full
Name          Type          Port Targ  Lun                Used by
-------------------------------------------------------------------
DISK100       disk             1    1    0
           RZ28    (C) DEC          0001
DISK210       disk             2    1    0                   D200
          SEAGATE  ST15230N         0005
        Switches:
          NOTRANSPORTABLE
        Size: 8356756 blocks
```

2. Invoke the HSUTIL utility and select the code load function by entering option "2" at the prompt:

```
HSZ20> run hsutil
  *** Available functions are:
    1. FORMAT
    2. CODE_LOAD
    3. EXIT

Enter the number of the function you wish to perform (1:3) [3] ?2
```

3. The program displays the single-device units available for code loading:

   Single device units on this controller include:

```
    Unit   Associated Device   SCSI Product ID
    200    DISK210             ST15230N
```

4. Select DISK210 as the source disk:

```
Which unit is the code to be loaded FROM ? 200
```

5. Because you have previously copied the firmware image at LBN 0 or the source disk, you can accept the default starting LBN of the device firmware image by pressing Return at the prompt:

```
What is the starting LBN of the code on the unit where the code is
to be loaded FROM [0] ? Return
```

6. Enter the product ID of DISK110. Enter this information exactly as it was reported by the program earlier, including any spaces or special characters: (You may, however, use only as many characters as are necessary to uniquely identify the device, as shown in the following example.)

```
What is the SCSI PRODUCT ID of the disk that you want code load TO
? RZ28
```

7. The program reports DISK110 as the only unattached device with the desired product ID:

```
Unattached devices on this controller include:
 Device    SCSI Product ID    Current Device Rev
 DISK110   RZ28     (C) DEC   0001
```

8. Select DISK110 as the target device:

```
Which device is the code to be loaded TO ? disk110
```

9. The program displays a series of informational messages: Pay particular attention to the warning messages:

```
^Y and ^C will be disabled while the code load operation is in
progress.

CAUTION:
     Loading the incorrect microcode could disable the destination
     device. If a failure occurs while loading FLASH memory, the
     destination device could be disabled. A backup of the disk
     drive should be done before a code load.

CAUTION:
     With some device code releases, a format is required after
     the device code load in order to make the device usable.
     Check the release notes of the device code or contact the
     device vendor to see if this is the case. Note: a format will
     destroy all data on the device.

CAUTION:
     In order to minimize the possibility of a SCSI bus reset,
     which could disable the destination device, it is recommended
     that you prevent IO operations to all other devices on the
     same port as the destination device.

After you answer the next question, the code load will start.
```

10. The program offers you the option of exiting:

```
Do you want to continue(y/n)[n] ?y
```

11. The program begins the device code load operation, finishing within two minutes:

```
HSUTIL Started at: 13-JAN-1946 05:51:47

Device code has been successfully downloaded to DISK110

HSUTIL - Normal Termination at: 13-JAN-1946 05:53:03
```

12. Enter the SHOW DISKxxx command for DISK110 to verify that the device code load operation was successful. Note that the program reports the firmware revision level as “0002”. The device is now available for use with its new firmware.

```
HSZ20> sho disk110

Name          Type          Port Targ  Lun                Used by
-----------------------------------------------------------------
DISK110       disk             1    1    0
           RZ28    (C) DEC          0002
```

6.3.3 Loading Code on a Supported Device (Format Required)

The following example demonstrates the use of the HSUTIL utility to download device firmware. In this case, the device’s product documentation specifies that a format operation is required after the code load. The example changes the device firmware in device DISK110 from revision 0001 to 0002. The source disk is DISK210.

The details of copying the device firmware image from a location external to the controller to the source disk are specific to the host operating system. In this example, the firmware was previously copied from the host to LBN 0 of DISK210.

1. Use the SHOW DEVICE command to report the available devices and the firmware level of each. Note that the firmware level of DISK110 is reported as “0001” and that it is an unattached device:

```
HSZ20> sho dev full
Name          Type          Port Targ  Lun                 Used by
------------------------------------------------------------------

DISK110       disk             1    1    0
           ST32550N               0001
        Switches:
          NOTRANSPORTABLE
        Size: 8356756 blocks
DISK210       disk             2    1    0                    D200
          SEAGATE  ST15230N       0005
        Switches:
          NOTRANSPORTABLE
        Size: 8356756 blocks
```

2. Invoke the HSUTIL utility and select the code load function by entering option “2” at the prompt:

```
HSZ20> run hsutil
  *** Available functions are:
    1. FORMAT
    2. CODE_LOAD
    3. EXIT

Enter the number of the function you wish to perform (1:3) [3] ?2
```

3. The program displays the unattached devices available for code loading:

```
Single device units on this controller include:
    Unit   Associated Device   SCSI Product ID
    200    DISK210             ST15230N
```

4. Select DISK210 as the source disk, by entering its unit number, “200”:

```
Which unit is the code to be loaded FROM ? 200
```

5. Because you have previously copied the firmware image to LBN 0 of the source disk, you can accept the default starting LBN of the device firmware image by pressing Return at the prompt:

```
What is the starting LBN of the code on the unit where the code is
to be loaded FROM [0] ? Return
```

6. Enter the product ID of DISK110. Enter this information exactly as it was reported by the program earlier, including any spaces or special characters. (You may, however, use only as many characters as are necessary to uniquely identify the device.)

```
What is the SCSI PRODUCT ID of the disk that you want code load TO
? ST32550N
```

7. The program reports DISK110 as the only single device unit with the desired product ID:

```
Unattached devices on this controller include:
    Device    SCSI Product ID     Current Device Rev
    DISK110   ST32550N            0001
```

8. Select DISK110 as the target device:

```
Which device is the code to be loaded TO ? disk110
```

9. The program displays a series of informational messages. Pay particular attention to the warning messages:

```
^Y and ^C will be disabled while the code load operation is in
progress.

CAUTION:
     Loading the incorrect microcode could disable the destination
     device.
     If a failure occurs while loading FLASH memory, the
     destination device could be disabled. A backup of the disk
     drive should be done before a code load.

CAUTION:
     With some device code releases, a format is required after
     the device code load in order to make the device usable.
     Check the release notes of the device code or contact the
     device vendor to see if this is the case.
     Note: a format will destroy all data on the device.

CAUTION:
     In order to minimize the possibility of a SCSI bus reset,
     which could disable the destination device, it is recommended
     that you prevent IO operations to all other devices on the
     same port as the destination device.
     After you answer the next question, the code load will start.
```

10. The program offers you the option of exiting:

```
Do you want to continue(y/n)[n] ?y
```

11. The program begins the device code load operation, finishing a number of minutes later. In this case, the device is reported as not ready after the code load, because it must be formatted before it is usable. The sense key error shown is typical for a device that requires formatting after it is code loaded:

```
HSUTIL Started at: 13-JAN-1946 04:39:54

Device code has been successfully downloaded to DISK110.
The code load has been successful. However, a subsequent test unit
ready command has failed for the following reason:

Error at PTL 1.1.0. Sense key:06 ASC/Q:29/00

You probably need to format the device before it will be usable.
You can do this by invoking HSUTIL again and picking the format
function.
HSUTIL - Normal Termination at: 13-JAN-1946 04:41:06
```

12. Use the SHOW DEVICE command to report that DISK110 is unusable to the controller:

```
HSZ20> sho disk110
Name          Type          Port Targ  Lun                 Used by
------------------------------------------------------------------
DISK110       disk             1    1    0
       Misconfigured: No device at this PTL, please see user guide
```

13. Invoke the HSUTIL utility and select the format operation by entering "1" at the prompt:

```
HSZ20> run hsutil
  *** Available functions are:    1. FORMAT
    2. CODE_LOAD
    3. EXIT

Enter the number of the function you wish to perform (1:3) [3] ?1
```

15. The program reports the unattached units available for formatting:

```
Unattached units on this controller include:
 Device    SCSI Product ID     Current Device Rev
 DISK110   ST32550N            0002
 DISK210   ST15230N            0005
```

16. Select DISK110 as the device to be formatted:

```
Enter the device to format ?disk110
```

17. The program displays a number of informational messages. Pay particular attention to the warnings:

```
^Y and ^C will be disabled while the format operation is in
progress.
 Format can take up to 40 minutes.

CAUTION:
    When you format a device, it will destroy the data on the
    device. A backup of the device should have been done if the
    data is important.

NOTE:
    In order to minimize the possibility of a SCSI bus reset, it is
    recommended that you prevent IO operations to all other devices
    on the same port as the destination device. If a SCSI bus reset
    occurs, the format may be incomplete and you may have to re-
    invoke HSUTIL.

After you answer the next question, the format will start
```

18. The program offers you the option of exiting before the format operation begins:

```
Do you want to continue(y/n)[n] ?y
```

19. The program begins the format operation, finishing a number of minutes later. It also informs you of the steps you need to take to make the device usable:

```
HSUTIL Started at: 13-JAN-1946 04:49:48
Format of DISK110 has successfully finished.
HSUTIL - Normal Termination at: 13-JAN-1946 05:22:33
```

20. Use the SHOW DEVICE command to report the available devices and the firmware level of each. Note that the firmware level of DISK110 is now reported as “0002”. The device is now available for use with its new firmware code:

```
HSZ20> sho dev full
Name          Type          Port Targ  Lun                  Used by
-------------------------------------------------------------------
DISK110       disk            1    1    0
          ST32550N                 0002
        Switches:
          NOTRANSPORTABLE
        Size: 8356756 blocks
DISK210       disk            2    1    0                     D200
          SEAGATE  ST15230N        0005
        Switches:
          NOTRANSPORTABLE
        Size: 8356756 blocks
```

6.3.4 Loading Code On an Unsupported Device

CAUTION

HSUTIL has been tested to perform the code load operation properly only with supported devices. Use HSUTIL to code load unsupported devices at your own risk.

Device compatibility with firmware code loading using the SCSI Write Buffer command varies greatly from vendor to vendor. While HSUTIL does check to see if a target disk is an unsupported device, it does not prevent you from continuing the code load operation on the device. You should understand that an unsupported device may be rendered unusable until it is factory-initialized, if the code load is unsuccessful.

The following example demonstrates the use of the HSUTIL utility to download device firmware to an unsupported device that does not require a format after its firmware is updated. The example changes the device firmware in device DISK110 from revision 0001 to 0002. The source disk is DISK210.

The details of copying the device firmware image from a location external to the controller to the source disk are specific to the host operating system. In this example, the firmware was previously copied from the host to LBN 0 of DISK210.

1. Use the SHOW DEVICE command to report the available devices and the firmware level of each. Note that the firmware level of DISK110 is reported as “0001”:

```
HSZ20> sho dev full
Name          Type          Port Targ  Lun                  Used by
-------------------------------------------------------------------

DISK110       disk            1    1    0
           RZ26    (C) DEC         0001
DISK210       disk            2    1    0                     D200
          SEAGATE  ST15230N        0005
        Switches:
          NOTRANSPORTABLE
        Size: 8356756 blocks
```

2. Invoke the HSUTIL utility and select the code load function by entering option "2" at the prompt:

```
HSZ20> run hsutil
  *** Available functions are:
    1. FORMAT
    2. CODE_LOAD
    3. EXIT

Enter the number of the function you wish to perform (1:3) [3] ?2
```

3. The program displays the single-device units available for code loading:

```
Available single device units on this controller include:
    Unit   Associated Device   SCSI Product ID
    200    DISK210             ST15230N
```

4. Select DISK210 as the source disk, by entering its unit number, "200":

```
Which unit is the code to be loaded FROM ? 200
```

5. Because you have previously copied the firmware image to LBN 0 of the source disk, you can accept the default starting LBN of the device firmware image by pressing Return at the prompt:

```
What is the starting LBN of the code on the unit where the code is
to be loaded FROM [0] ? Return
```

6. Enter the product ID of DISK110. Enter this information exactly as it was reported by the program earlier, including any spaces or special characters. (You may, however, use only as many characters as are necessary to uniquely identify the device, as shown in the following example.)

```
What is the SCSI PRODUCT ID of the disk that you want code load TO
? RZ26
```

7. The program reports DISK110 as the only unattached device with the desired product ID:

```
Unattached devices on this controller include:
 Device     SCSI Product ID     Current Device Rev
 DISK110    RZ26     (C) DEC    0001
```

8. Select DISK110 as the target device:

```
Which device is the code to be loaded TO ? disk110
```

9. HSUTIL compares the device's SCSI product ID with those in its table of supported devices. In this case, the program displays a warning, because DISK110 is not a supported device. The program gives you the option of exiting. Enter "y" to continue:

```
RZ26     (C) DEC is not a supported device.
Proceed at your own risk.
Do you want to continue(y/n) [n] ?y
```

CAUTION

Be particularly careful to enter the correct information for your device. If you enter an incorrect parameter, you may render the device unusable until it is factory-initialized.

> **NOTE**
>
> Because the device is unsupported, you must configure the program for the download process by entering several device parameters. Refer to your device release notes and product documentation to locate the information required to respond to the following prompts.
>
> Do not use the parameters entered in this example. They are presented for demonstration purposes only. Use the parameters appropriate for your device, as specified in your device's product documentation.
>
> Some or all of the following parameter queries may be displayed, depending upon the device model.

10. The first parameter required is the size of the firmware image. In this case, you accept the default value of 512 blocks, by pressing the Return key:

    ```
    What is the TOTAL SIZE of the code image in 512 byte blocks [512]
    ? Return
    ```

11. In this case, you tell the program that the firmware is to be transferred in multiple SCSI Write Buffer commands, by entering "n" at the prompt:

    ```
    Should the code be downloaded with a single write buffer command
    (y/n) [y] ? n
    ```

12. Tell the program how many 1024-byte blocks are to be transferred by each Write Buffer command. The default amount is "8", specifying a write buffer size of 8 x 1024 = 8192 bytes. In this case, select the default value by entering "8":

```
What BUFFER SIZE, module 1024, does the drive require (2,4,8,16,32)
   [8] ? 8
```

13. Tell the program whether or not the device only supports the SCSI Write Buffer Download Microcode and Save mode. In this case, the device does support this mode, and you press the Return key to accept the default of "y":

    ```
    Does the target device support only the download microcode and
    SAVE  (y/n) [y] ? Return
    ```

14. Tell the program whether or not the device requires the firmware to be transferred with the byte order reversed. In this case, the device requires normal byte order, and you press the Return key to accept the default of "n":

    ```
    Would you like the bytes REVERSED (few devices require this) (y/n)
    [n] ? Return
    ```

15. The program is now configured for your unsupported device, and it displays a series of informational messages. Pay particular attention to the warning messages

    ```
    ^Y and ^C will be disabled while the code load operation is in
    progress.

    CAUTION:
         Loading the incorrect microcode could disable the destination
         device. If a failure occurs while loading FLASH memory, the
         destination device could be disabled. A backup of the disk
         drive should be done before a code load.
    ```

```
CAUTION:
    With some device code releases, a format is required after
    the device code load in order to make the device usable.
    Check the release notes of the device code or contact the
    device vendor to see if this is the case. Note: a format will
    destroy all data on the device.
CAUTION:
    In order to minimize the possibility of a SCSI bus reset,
    which could disable the destination device, it is recommended
    that you prevent IO operations to all other devices on the
    same port as the destination device.
After you answer the next question, the code load will start.
```

16. The program offers you the option of exiting:

CAUTION

If you have any doubt that you have configured the program properly for your device, exit the program now. If you continue the code load operation with incorrect parameters, you may render the device unusable until it is factory-initialized.

```
Do you want to continue(y/n)[n] ?y
```

17. The program begins the device code load operation, finishing a number of minutes later:

```
HSUTIL Started at: 13-JAN-1946 05:51:47
Device code has been successfully downloaded to DISK110
HSUTIL - Normal Termination at: 13-JAN-1946 05:53:03
```

18. Enter the SHOW DISKxxx command for DISK110, to verify that the device code load operation was successful. Note that the program reports the firmware revision level as “0002”. The device is now available for use with its new firmware.

```
HSZ20> sho disk110
Name          Type          Port Targ  Lun                Used by
------------------------------------------------------------------
DISK110       disk             1    1    0
           RZ26    (C) DEC          0002
```

19. If your unsupported device’s product documentation specifies that the device requires a format operation after a code load operation, you must perform it at this point. Refer to Section 6.3.3 for an example of a device code load with subsequent format operation.

6.3.5 Sample Host Copy Script

The details of copying the firmware image to the source disk from the host disk are specific to the host operating system. Following is a sample script, for use under the Digital UNIX™ operating system:

```
# disklabel -z /dev/rrza40c
# dd if=rz28_t436a_dec2104.fup of=/dev/rrza40a bs=512
512+0 records in
512+0 records out
```

Note that, if the disklabel is not set to zero, Digital UNIX attempts to protect the disklabel’s location (LBN 0.) You must zero the disklabel to the source disk before performing the code load operation. Failure to do this results in the following error:

```
# dd if=rz28_t436a_dec2104.fup of=/dev/rrza40a bs=512
dd write error: Read-only file system
8+0 records in
0+0 records out
```

6.3.6 Abort Codes

In the event that HSUTIL terminates before it successfully completes a format or code load operation, it reports one of the abort codes in Table 6–1.

Table 6–1 Abort Codes

Abort Code	Description
1	FAO returned either FAO_BAD_FORMAT or FAO_OVERFLOW
2	Bad return from TS$READ_TERMINAL_DATA
3	TS$READ_TERMINAL_DATA returned either an ABORTED or INVALID_BYTE_COUNT
4	User requested an abort via ^Y or ^C
5	An error occurred on a SCSI command
6	Can’t find the pub, device is probably missing

6.3.7 HSUTIL Messages

Following is a list of status and prompt messages displayed by the HSUTIL utility:

Message:

`Insufficient resources`

Explanation: HSUTIL cannot find perform the operation because internal controller resources are not available.

Message:

`HSUTIL - Normal Termination at:` *`time`*

Explanation: HSUTIL finished the previous operation without error, at the time indicated.

Message:

`Unable to change operation mode to maintenance for unit`
`unit_number`

Explanation: HSUTIL was unable to put the source unit into maintenance mode to enable formatting or code load.

Message:

`Unit` *`unit_number`* `does not exist`

Explanation: HSUTIL cannot locate the unit indicated in the operation requested.

Message:

`Unit` *`unit_number`* `successfully allocated`

Explanation: HSUTIL has allocated the unit for code load operation. At this point, the unit and its associated device are not available for other subsystem operations.

Message:

`Unable to allocate unit`

Explanation: The unit could not be allocated. An accompanying message explains the reason.

Message:

`Unit` *`unit_number`* `does not have any media present`

Explanation: There is no media in the device. (For devices with removable media only.)

Message:

`No available units`

Explanation: There are no on-line, single-device units in the system.

Message:

`Unit is owned by another sysap`

Explanation: Device cannot be allocated because it is being used by another subsystem function or local program.

Message:

`Disk unit` *`unit_number`* `is not valid.`

Explanation: The device unit number specified by the user is not a legal unit number or does not exist.

Message:

`HSUTIL termination requested.`

Explanation: The user has terminated HSUTIL with a key command.

Message:

`Unit` *`unit_number`* `is in maintenance mode`

Explanation: Device cannot be formatted or code loaded because it is being used by another subsystem function or local program.

Message:

`Unit` *`unit_number`* `is not operational`

Explanation: The device shown appears broken or missing to the controller.

Message:

`Unable to allocate memory`

Explanation: The controller was unable to furnish HSUTIL with sufficient memory to perform the operation.

Message:

`device_name` `is not a supported device. Proceed at your own risk`

Explanation: HSUTIL has determined that the device in the operation is not supported. If you proceed with the operation, damage to the device may result.

Message:

```
Format can take up to xxx minutes.
```

Explanation: Self-explanatory.

Message:

```
^Y and ^C will be disabled while the format operation is in
progress.
```

Explanation: You can terminate the format operation by entering either keystroke combination at any time before the program begins to issue SCSI commands.

Message:

```
^Y and ^C will be disabled while the code load operation is in
progress.
```

Explanation: You can terminate the code load operation by entering either keystroke combination at any time before the program begins to issue SCSI commands.

Message:

```
Single device units on this controller include:
Unit   Associated Device   SCSI Product ID
```

Explanation: The program lists the available, single-device units, their associated devices and their SCSI product IDs.

Message:

```
After you answer the next question, the code load will start.
```

Explanation: Self-explanatory.

Message:

```
CAUTION:

 Loading the incorrect microcode could disable the destination
device. If a failure occurs while loading FLASH memory, the
destination device could be disabled. A backup of the disk drive
should be done before a code load.
```

Explanation: Normal code load operations do not affect the data on a disk device. To avoid the loss of your data in the event of an unexpected situation, you should back up any data on a storage device before proceeding with a code load operation to it.

Message:

```
CAUTION:
In order to minimize the possibility of a SCSI bus reset, which
could disable the destination device, it is recommended that you
prevent IO operations to all other devices on the same port as the
destination device.
```

Explanation: (Displayed in code load only.) A SCSI bus reset can occur if the controller is manually rebooted or if it detects an error during normal subsystem operation. The more active devices there are on the same port as the target device, the greater the chance that an error causing a SCSI bus reset may occur. By minimizing the level of activity on the device port being used for code loading, the user minimizes the chances of a SCSI bus reset that could render a target device unusable.

Message:

```
CAUTION:

In order to minimize the possibility of a SCSI bus reset, it is
recommended that you prevent IO operations to all other devices on
the same port as the destination device.  If a SCSI bus reset
occurs, the format may be incomplete and you may have to re-invoke
HSUTIL.
```

Explanation: (Displayed in format only.) A SCSI bus reset can occur if the controller is manually rebooted or if it detects an error during normal subsystem operation. The more active devices there are on the same port as the target device, the greater the chance that an error causing a SCSI bus reset may occur. By minimizing the level of activity on the device port being used for formatting, the user minimizes the chances of a SCSI bus reset that could render a target device unusable.

Message:

```
CAUTION:
When you format a device, it will destroy the data on the device.
A backup of the device should have been done if the data is
important.
```

Explanation: The SCSI format operation is a low-level process that destroys all the data on the disk. To avoid losing your data, you should back it up before proceeding with the format operation.

Message:

```
After you answer the next question, the format will start
```

Explanation: Self-explanatory.

Message:

```
The code load has been successful. However, a subsequent test unit
ready command has failed for the following reason:

Error at PTL ptl_address. Sense key: sense_key  ASC/Q: asc/q

You probably need to format the device before it will be usable.
You can do this by invoking HSUTIL again and picking the format
function. Do not delete the unit or device until after the HSUTIL
format function has finished.
```

Explanation: Some device firmware releases require the device to be formatted after device firmware is downloaded. This message indicates that SCSI Test Unit Ready commands, executed after a successful code load, failed. The most likely cause of the failure is that the device must be formatted before it can be used.

Message:

```
WARNING:
With some device code releases, a format is required after the
device code load in order to make the device usable. Check the
release notes of the device code or contact the device vendor to
see if this is the case. Note: a format will destroy all data on
the device.
```

Explanation: Self-explanatory.

Message:

```
HSUTIL aborted, code code_number
```

Explanation: HSUTIL terminated with an error. The abort code number is shown. See Table 6–1 for a description of HSUTIL abort codes.

Message:

```
HSUTIL Started at: time
```

Explanation: Self-explanatory.

Message:

```
Exclusive access is declared for unit unit_number
```

Explanation: Another subsystem function has reserved the unit shown.

Message:

```
This unit is marked inoperative
```

Explanation: The device has failed and has been tagged as such.

Message:

```
The RUNSTOP_SWITCH is set to RUN_DISABLED for unit unit_number
```

Explanation: The RUN\NORUN unit indicator for the unit shown is set to NORUN. The disk is not spun up.

Message:

```
Device code has been successfully downloaded to device_name
```

Explanation: The code load operation to the device shown was completed without error.

Message:

```
Unattached devices on this controller include:
 Device    SCSI Product ID      Current Device Rev
```

Explanation: The program lists the unattached devices, their SCSI product IDs, and their firmware revision levels.

Message:

```
No available unattached devices.
```

Explanation: The program could find no unattached devices to list.

Message:

```
Do you want to continue(y/n)[n] ?
```

Explanation: Self-explanatory.

Message:

```
Which device is the code to be loaded TO ?
```

Explanation: Enter the device_name of the target disk.

Message:

```
Which unit is the code to be loaded FROM ?
```

Explanation: Enter the unit number of the source disk. Make sure you use the device's *unit* number, and not its *disk* number.

Message:

```
What is the SCSI PRODUCT ID of the code load destination device ?
```

Explanation: Enter the product ID, as reported in the SHOW DEVICE command. Enter all alpha and numeric characters.

Message:

```
The format time for this device is unknown.  It could take from 15
to 100 minutes.
```

Explanation: Because this is an unsupported device, the format time for this device could not be found in program's internal table of format times.

Message:

Unit *unit_number* is not ONLINE

Explanation: The unit cannot be accessed by the program.

Message:

```
Enter a device to format ?
```

Explanation: Enter the device_name of the target disk.

Message:

```
What is the starting LBN of the code on the unit where the code is
to be loaded FROM [0] ?
```

Explanation: If the starting LBN of the device firmware image on the source disk is other than 0, enter it as a numeric string. In most cases, the default LBN of "0" is used. It can be entered by pressing Return.

Message:

```
What BUFFER SIZE, module 1024, does the drive require
(2,4,8,16,32) [8] ?
```

Explanation: This message is displayed if HSUTIL detects that an unsupported device has been selected as the target device and if the user elects to download the firmware image using more than one SCSI Write Buffer command. The user must specify the number of bytes to be sent in each Write Buffer command. The default buffer size is 8192 bytes. A firmware image of 256 KB, for example, can be code loaded in 32 Write Buffer commands, each transferring 8192 bytes. In this example, the correct entry for the buffer size would be "8".

Message:

```
What is the TOTAL SIZE of the code image in 512 byte blocks [512]
?
```

Explanation: This message is displayed if HSUTIL detects that an unsupported device has been selected as the target device. The user must enter the total number of 512-byte blocks of data to be sent in the code load operation. For example, a firmware image that is 262,144 bytes long would require 512, 512-byte blocks.

Message:

```
Does the target device support only the download microcode and
SAVE (y/n) [y] ?
```

Explanation: This message is displayed if HSUTIL detects that an unsupported device has been selected as the target device. The user must specify whether of not the device supports the SCSI Write Buffer command's Download And Save function.

Message:

```
Would you like the bytes REVERSED (few devices require this) (y/n)
[n] ?
```

Explanation: Some devices require that the bytes be reversed when downloading the firmware image. The user must specify whether the device requires this action.

Message:

```
Should the code be downloaded with a single write buffer command
(y/n) [y] ?
```

Explanation: : This message is displayed if HSUTIL detects that an unsupported device has been selected as the target device. The user must tell the program whether to download the firmware image to the device in one or more contiguous blocks, each corresponding to one SCSI Write Buffer command.

Message:

```
*** Available functions are:
    1. FORMAT
    2. CODE_LOAD
    3. EXIT

Enter the number of the function you wish to perform (1:3) [3] ?
```

Explanation: Select either the format or code load functions by entering a menu item number.

6.4 Changing Volume Serial Numbers

Use the Change Volume Serial Number Utility (CHVSN) to view and change storage device volume serial numbers.

The following example demonstrates the use of the CHVSN utility to change the volume serial number of a device. The example changes the volume serial number of the device at PTL 110 from 00000000 00000000 to 00000000 00000001:

1. Invoke the CHVSN program to view the volume serial number of the devices at PTL 110.

```
HSZ20> RUN CHVSN
Device (port target lun)  [EXIT] ? 1 1 0
CHVSN: Volume Serial Number is 00000000 00000000
```

2. Change the volume serial number for this volume by entering a new number:

```
Update CHVSN (Y/N) [N] ? Y
CHVSN: Volume Serial Number is 00000000 00000001
```

3. Exit the CHVSN program:

```
Device (port target lun)  [EXIT] ? Return
CHVSN - Normal Termination
```

7

Troubleshooting

This chapter provides miscellaneous information you may need to correct subsystem performance problems.

7.1 Troubleshooting Tips

Following are some tips you can use when troubleshooting controller and subsystem problems:

1. Make a note of all visual indicators (OCP, device LEDs, EMU, or error messages) available to you.
2. Extract and read host error logs.
3. Errors can be intermittent. Reset the controller to see if the error clears. (Record which devices have lit/flashing fault LEDs before resetting, as a reset may temporarily clear the LED even though the fault remains.)
4. See if the error indication changes after resetting the controller. If the error remains the same, look up information for that error. If the indication changes, look up information for the newer error.
5. Always consider reseating the controller and/or cache module when troubleshooting. Poor connections between module and backplane can cause a variety of errors.

CAUTION

Do not use a controller failure situation as an opportunity to move devices or otherwise reconfigure your subsystem. Doing so will prevent the controller from communicating with its units once the fault is corrected.

7.2 Direct Fault Isolation

The easiest way to isolate a failed controller or subsystem component is to use *direct* fault isolation information, provided from the system. You have four primary sources for direct fault isolation information:

- OCP codes during initialization—Controller problems that show up in the initialization diagnostics provide you with a direct indication of the failed component. The maintenance actions specified in the RAID Array 310 hardware guide indicate directly which component has failed and should be replaced.
- OCP codes during CLI-initiated self test—If you put a controller in the self-test mode by issuing the SELFTEST THIS/OTHER_CONTROLLER command, the OCP codes in the RAID Array 310 hardware guide indicate directly which component has failed and should be replaced.

- CLI messages—Many CLI error messages include error codes associated with component failures. You can use this error code information to isolate failed components. See Section 7.3 for further information on fault isolation using error codes.
- Host error logs—Host error logs typically display SCSI sense data that provides a very high level of resolution on a problem. In many cases, host error logs automatically interpret the sense data to indicate which components have failed.

Remember that you may have to interpret the failed component information from these sources, depending upon your controller model.

7.3 Isolating Faults through CLI Messages

The firmware on the maintenance terminal reports two basic types of error messages:

- Stored messages—The controller sends stored messages to your maintenance terminal in response to some action you have performed, such as running a utility or entering a CLI command.
- Spontaneous messages—The controller sends spontaneous status and error messages to your maintenance terminal in response to unusual subsystem conditions.

The following sections describe these messages.

7.3.1 Stored Messages

Stored messages are sent in response to some operator action. The messages sent are usually associated with a utility program being run or a CLI command being entered. For a listing of stored messages see Section A.2 CLI Messages in Appendix A.

7.3.2 Spontaneous Messages

Spontaneous messages are part of a CLI error report (CER) system. The CER system causes the message text to appear on a maintenance terminal at any time, in response to some subsystem action or condition.

Spontaneous messages are displayed only if all of the following conditions are true:

- A maintenance terminal is connected to the controller's maintenance terminal port.
- The subsystem has finished initializing.
- You are not currently running a utility program on the maintenance terminal.
- The maintenance terminal is not actively displaying input from another source, such as event logging or last failure logging.
- No CLI commands are in progress on maintenance terminal.
- The fault management utility (FMU) has not been set to disable error messages.

7.3.3 Error Message Review

If any of the required conditions for message display is not true, the spontaneous message display does not occur. In this case, the firmware stores the messages. It displays the 15 most recent event messages when a maintenance terminal is connected and you press the Return key.

Often, error message review continues to occur each time Return is pressed. To clear the terminal of the errors, enter the CLEAR_ERRORS CLI command. (You may want to make a note of the errors before clearing them, because you cannot recall them afterwards.)

You can use FMU to review error messages and to configure the error reporting firmware to report in specific ways. See Section 5.1: Viewing Error Logs in Chapter 5 for further information on the FMU utility.

7.3.4 Message Listing

The following section lists spontaneous messages you may encounter. The controller sends these messages when the specific fault is detected, regardless of whether or not you are interactively viewing or using the maintenance terminal. These messages differ in this respect from the ones listed in Chapter 3 which appear based on your interactive use of the CLI. The CER messages are presented in alphabetic order to make them easy to locate, but they cover the following error categories:

- Configuration and CLI
- Last Failure Entry
- System Information Structure and System Information Page
- Diagnostic and initialization
- NVPM
- Read cache
- Shelf
- Write-back cache
- Interactive CLI messages

Consult your firmware release notes for updates to the list of event messages.

Message

`All NVPM components initialized to their default settings`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`Both HS`*`xxx`*` controllers are using SCSI address 6`

Explanation: There is a hardware problem with the HA000 controller shelf. This problem probably involves the shelf backplane.

Message

`Both HSxxx controllers are using SCSI address 7`

Explanation: There is a hardware problem with the HA000 controller shelf. This problem probably involves the shelf backplane.

Message

`Cache battery charge is low`

Explanation: The write-back cache battery is partially discharged. Any unwritten cache data on the cache is flushed. Non-RAIDset units will be accessed in read cache mode. RAIDsets will be unavailable. Replace the cache battery.

Message

`Cache battery failed diagnostic testing`

Explanation: The write-back cache battery has failed the diagnostic tests. Any unwritten cache data on the cache is flushed. Non-RAIDset units will be accessed in read cache mode. RAIDsets will be unavailable. Replace the cache battery.

Message

`Cache module failed diagnostic testing - half not accessible`

Explanation: This message can appear for either read cache or write-back cache. Up to 50 percent of the cache memory has failed the diagnostic tests. If the controller has a write-back cache, unwritten data in the cache module is lost. Replace the cache module.

Message

`Cache module failed diagnostic testing`

Explanation: This message can appear for either read cache or write-back cache. The cache has failed the diagnostic tests. If the controller has a write-back cache, unwritten data in the cache module is lost. Replace the cache module.

Message

`Cache module has metadata incompatible with this firmware`

Explanation: The subsystem was not properly run down before changing firmware versions. There may be unwritten cache data which cannot be recovered, because the cache metadata format has changed along with the firmware. Correct the problem in one of two ways:

- Restore the previous firmware version, and properly run down the subsystem.
- Enter the CLI command CLEAR_ERRORS INVALID_CACHE, which allows you to use the cache module (although you will lose the unwritten cache data).

Message

```
Configuration information deleted due to internal
inconsistencies
```

Explanation: This message is displayed if a test of nonvolatile memory shows corruption. The configuration information for the controller is deleted when this message is displayed.

Message

```
Controller Characteristics component reformat failed during
NVPM Revision Level 1 to 2 reformat
```

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

```
Controllers misconfigured. Type SHOW THIS_CONTROLLER
```

Explanation: If this message appears, examine the SHOW THIS_CONTROLLER display to determine the source of the misconfiguration.

Message

```
Controller shelf fixed
```

Explanation: The controller shelf has been correctly repaired.

Message

```
Controller shelf has a bad power supply or fan
```

Explanation: Troubleshoot the system to isolate and replace the failed component.

Message

```
Device and/or Storageset names changed to avoid conflicts
```

Explanation: Digital adds new CLI keywords at each new firmware release that can conflict with existing device and/or storageset names. When this happens, firmware changes your device and/or storageset names and sends this message. The functional operation of your configuration is not changed when this message appears.

Message

```
Disk mirroring option has been turned on without a valid
license key
```

Explanation: Disk mirroring is a licensed feature. This message indicates that it has been turned on without entering a valid license key in the firmware Licensing System (FLU).

Message

`EVL control block allocation failed`

Explanation: The controller was unable to allocate enough memory to run the EVL utility. Report this error as a subsystem problem.

Message

`FMU control block allocation failed`

Explanation: The controller was unable to allocate enough memory to run the FMU utility. Report this error as a subsystem problem.

Message

`Host Access Disabled`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`HOST port FAILED Diagnostics`

Explanation: The host port of the controller has failed diagnostics. See **Error! Reference source not found.** for more information on diagnostics.

Message

```
Invalid cache – CLI command set reduced until INVALID_CACHE
error is cleared
```

Explanation: Data in the cache does not match the controller image of cache data. Use the CLEAR_ERRORS INVALIDC_CACHE command to delete the cache data.

Message

`Last Failure Entry:` *`n`* `reset – power-on time reset`

Explanation: The power-on time contained in the Last Failure Entry identified by *n*was greater than the current power on time. That condition renders the identified Last Failure Entry unusable, resulting in its being reset to default settings. Reset of Last Failure Entry should be reported as a subsystem problem.

Message

`Last Failure Entry:` *`n`* `reset – predecessor invalid`

Explanation: The Last Failure Entry that immediately precedes the Last Failure Entry identified by *n* was found to be invalid. That condition renders the identified Last Failure Entry unusable, resulting in its being reset to default settings. Reset of Last Failure Entry should be reported as a subsystem problem.

Message

`Last Failure Entry: `*`n`*` reset – predecessor reset`

Explanation: One or more of the Last Failure Entries that immediately precede the Last Failure Entry identified by *n* was reset to default settings. That condition renders the identified Last Failure Entry unusable, resulting in its being reset to default settings. Reset of Last Failure Entry should be reported as a subsystem problem.

Message

`Last Failure Entry: `*`n`*` reset – previously invalidated`

Explanation: During termination of controller operation processing, the Last Failure Entry identified by *n* was reset to default settings for one or more of the reasons listed above. In this case, the reason for resetting the entry is not retained. Reset of Last Failure Entry should be reported as a subsystem problem.

Message

`Last Failure Entry: `*`n`*` reset – revision mismatch`

Explanation: The revision level contained in the Last Failure Entry identified by *n* did not match the revision level understood by the current HSOF firmware. The identified Last Failure Entry is therefore not translatable and was reset to default settings. Reset of Last Failure Entry should be reported as a subsystem problem.

Message

`Local Terminal Port FAILED Diagnostics`

Explanation: The maintenance (EIA–423) terminal port has failed diagnostics. See Chapter 7, Section 7.3 Exercising the Subsystem for more information on diagnostics.

Message

```
Nonvolatile parameter/Write Journal Memory backup battery has
failed.
```

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`NVPM `*`component-name`*` component initialized to default settings`

Explanation: The NVPM component named by *component-name* has been reinitialized. Performing a controller reset may only clear this error message until the next time the controller is reset because the error could be caused by a fault in NVPM itself. If this error persists, replace the controller module.

Message

`NVPM cache information component initialized to default settings`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`NVPM Controller Characteristics component initialized to default settings`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`NVPM FMU Parameters component initialized to default settings`

Explanation: FMU settings have been lost. Default FMU options will be in place until you run FMU to change them. If the error persists, replace the controller module.

Message

`NVPM Product Information component initialized to default settings`

Explanation: The controller product identification setting has been reset. If the controller "name" is not what it used to be (for example, `HSZ20`) replace the controller module immediately.

Message

`NVPM Recursive Bugcheck Information component initialized to default settings`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`NVPM Revision level updated from` *n toN*

Explanation: The format of the NVPM has changed as a result of installing a newer program card (containing updated firmware). However, all subsystem configuration information has been retained.

Message

`NVPM System Information Page component initialized to default settings`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`NVPM User Interface Parameters component initialized to default settings`

Explanation: Terminal setting information has been lost. To correct this problem, enter the SHOW THIS_CONTROLLER and SHOW OTHER_CONTROLLER commands to determine the current terminal settings. Compare the terminal settings with the CONFIGURATION.INFO output information, and use the SET THIS_CONTROLLER and SET OTHER_CONTROLLER commands to restore terminal settings. If the error persists, replace the controller module.

Message

`NVPM Volume Serial Number component initialized to default settings`

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

`PCMCIA ECC invoked successfully. Replacement of PCMCIA card is recommended.`

Explanation: During controller initialization, the error correction code (ECC) detected and corrected an error in the firmware on the program card. The errors indicate that the card is becoming unreliable and should be replaced as soon as possible.

Message

`RAID5 option has been turned on without a valid license key`

Explanation: RAID5 is a licensed feature. This message indicates that it has been turned on without entering a valid license key in the firmware Licensing System (FLU).

Message

`Restart of the controller required to apply new patch`

Explanation: You used the Code Patch utility to enter a firmware patch, but the patch will not be applied until you restart the controller.

Message

`Restart of this controller required`

Explanation: A changed parameter requires reinitialization of this controller to take effect.

Message

`SCSI Device and HS`*`xxx`*` controller both configured at SCSI address 6`

Explanation: This message appears when a device is accidentally configured as SCSI ID 6, and two controllers (SCSI IDs 6 and 7) are in a dual-redundant configuration.

Message

`SCSI port` *`n`* `FAILED Diagnostics`

Explanation: A SCSI–2 port has failed diagnostics. This message can appear even if you do not have a host connection. The variable n indicates which port failed.

Message

`Serial number initialized due to format error`

Explanation: An invalid serial number was entered for the second controller of a dual-redundant pair.

Message

`Shelf` *`xx`* `fixed`

Explanation: Shelf number *xx* has been correctly repaired.

Message

`Shelf` *`xx`* `has a bad power supply or fan`

Explanation: Troubleshoot the system to isolate and replace the failed component.

Message

`SWAP signal cleared - all SWAP interrupts re-enabled`

Explanation: This message indicates that the swap signal is now cleared.

Message

```
System Information Page and Last Failure entries reset to
default settings.
```

Explanation: This message is reported in either of two cases:

- All structures contained in the SIP and the Last Failure entries have been reset to their default settings. This is a normal occurrence for the first time boot following manufacture of the controller module and during the transition from one firmware version to another if, and only if, the format of the SIP is different between the two versions.
- All structures contained in the SIP and the Last Failure entries have been reset to their default settings as the result of certain manufacturing configuration activities.
  Report this CER message as a subsystem problem.

Message

`System Information structure reset to default settings`

Explanation: The System Information structure within the SIP has been reset to default settings. The only known cause for this event is an I960 processor hang caused by an unimplemented memory region reference. Controller modules equipped with a watchdog timer will reboot within a few seconds. Controller modules without a watchdog timer will hang, as indicated by the OCP LEDs, and must be reset. Report this CER message as a subsystem problem.

Message

```
The following firmware Licensing Service component elements
were initialized to default settings: [ n ...
```

Explanation: Licensing information for licensed features has been lost. To correct this problem, reenter your license keys through FLS.

Message

```
The following NVPM Configuration Information component
elements\were initialized to default settings: [ n ...
```

Explanation: The settings given by *n* have been initialized in connection with another NVPM error. To clear this error, perform the following procedure:

1. Enter the following commands:
   CLI> **SHOW DEVICES**
   CLI> **SHOW UNITS**
   CLI> **SHOW STORAGESETS**

1. Compare the information displayed with a printout of the CONFIGURATION.INFO file or with a copy of the most current configuration.
2. Reconfigure the necessary devices, units, or storagesets.
   If the error persists, replace the controller module.

Message

```
The following NVPM Manufacturing Failure Information
component elements were initialized to default settings:  list
of component elements
```

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

```
This controller has a missing cache module
```

Explanation: The cache module is missing or is not seated properly. (Controller NVMEM indicates that a cache module is expected because there may be unwritten cache data.) You can either find/reseat the module, or enter the CLI command CLEAR_ERRORS INVALID_CACHE, which will allow you to run (in write-through mode) without a cache but without accessing RAIDsets.

Message

```
This controller has an invalid cache module
```

Explanation: The wrong cache module is present. This means the serial number stored in controller NVMEM and in the cache do not match, and unwritten cache data exists. (This message also can occur for a new, uninitialized module.) Correct the problem in one of two ways:

- Replace this cache with the correct one for this controller.
- Enter the CLI command CLEAR_ERRORS INVALID_CACHE, which allows you to use the random module.

Message

```
Unable to clear SWAP signal on shelf  xx - all SWAP interrupts
disabled
```

Explanation: The subsystem is unable to clear the swap signal for a swapped device, where *xx* is the shelf number. This could indicate an unsupported SBB or no power to the device shelf.

Message

```
Unknown NVPM Revision Level
```

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

```
Unknown reformat stage encountered during NVPM Revision Level
1 to 2 reformat
```

Explanation: Replace the controller immediately if this NVPM message occurs. Do not continue to use the controller.

Message

```
Write-back cache option has been turned on without a valid
license key
```

Explanation: Write-back cache is a licensed feature. This message indicates that it has been turned on without entering a valid license key in the firmware Licensing System (FLU).

7.4 Host Error Logging Tools

Events related to controller and device operation are usually recorded at the host in the form of an error log. Review the host error logs using the appropriate host error-translation program. Host error logs typically provide the greatest level of detail about error conditions associated with the controller and its devices.

Host error logs are operating-system specific. See the documentation for your host system for details on using a host error translation program to interpret host error logs.

7.5 Isolating Faults through Controller Error Codes

Your controller constantly monitors itself and the subsystem for unusual conditions. When an error condition is discovered, the controller captures information about the condition, and forms two basic types of error codes from them:

- Instance code—A unique, 8-digit, hexadecimal number that identifies an event or error detected. There is a unique instance code for each event or error known to the firmware. Instance codes are most often associated with events or errors that are recoverable, but they may also be reported in situations involving unrecoverable errors.

- Last failure code—A unique, 8-digit, hexadecimal number that identifies an unrecoverable condition detected. There is a unique last failure code for each unrecoverable condition known to the firmware. Last failure codes are always associated with error conditions that are unrecoverable.

7.5.1 Accessing Controller Error Codes

Your controller reports instance and last failure error codes to the maintenance terminal in the form of spontaneous error reports and user-initiated FMU reports.

If requested by the host, the controller also reports the codes via the host bus in the form of SCSI sense data.

7.5.1.1 Spontaneous Error Messages

The following example shows a typical spontaneous event logging error report.

```
%EVL--28-Jul-1994 07:44:48-- Instance Code: 01010302
```

Note that the instance code associated with this event is reported to aid you in fault isolation.

7.5.1.2 Viewing Error Logs

You can use the FMU SHOW LAST_FAILURE command to access error codes and other information on a recent error condition. Enter the command with the MOST_RECENT qualifier to get information on the most recent error. Use the ENTRY # qualifier to specify a particular error. See Chapter 6 for detailed information on viewing error logs.

An example of an FMU error report follows:

```
FMU> SHOW LAST_FAILURE MOST_RECENT
Last Failure Entry: 4 Flags: 0007FA80
%FMU-01-Last Failure Event, Instance Code: 016E2D02
 Power On Time: 0 Years, 41 Days, 4 Hours, 49 Minutes, 8
Seconds
 Controller Model: SC4200 Serial Number: ZG30355555 Hardware
Version: 0000(00)
 Controller Identifier:
  Unique Device Number: 000130355555 Model: 40(28) Class:
1(01)
 HSOF Version: V20(20)
 Informational Report
 Instance Code 016E2D02 Description:
  The CACHEA0 DRAB detected a Nonexistent Memory Error
condition during
  an I960 attempt to read CACHEA0 memory.
 Last Failure Code: 018900A0 (No Last Failure Parameters)
 Last Failure Code 018900A0 Description:
  A processor interrupt was generated with an indication that
a memory system.
  problem occurred.
```

Note that this report is associated with an error event that is unrecoverable and includes both an instance code and a last failure code.

7.5.2 Extracting Repair Action Codes From Error Codes

Each instance and error code contains a field of two numbers that indicates the specific repair action required to resolve the error condition, as shown in Table 7–1.

Table 7–1 Repair Action Code Field

					Repair Action Code Field			
Digit #	1	2	3	4	5	6	7	8
Instance Code Example	0	1	6	E	2	D	0	2
Last Failure Code Example	0	1	8	9	0	0	A	0

Note that the repair action code field for both instance and last failure codes consists of the 5th and 6th digits in the code. The two repair codes that are associated with the sample error log shown in Section 7.5.1.2 are “2D” and “00”.

Note that the repair action codes from the sample instance and last failure codes are not the same. You may find that there is more than one recommended repair action code in many error situations.

7.5.3 Using Repair Action Codes for Fault Isolation

Once you have extracted the repair action codes from the error codes, you can use Table 7–2 to determine what maintenance action to take to repair your controller.

To use the table, look up the repair action code in the left-hand column in the table and perform the repair action noted in the right-hand column. In some instances, you may be directed to another repair action code for more specific instructions. Follow this procedure for each repair action code you have for your controller error condition.

Some interpretation of the maintenance actions recommended may be necessary. Use other information you may have gathered from OCP error codes, maintenance terminal error messages, and host error reports to determine the likely cause of the problem.

Table 7–2 Repair Action Codes

Code	Description
00	No action necessary.
01	An unrecoverable hardware detected fault occurred or an unrecoverable firmware inconsistency was detected, proceed with controller support avenues. Contact Digital Services.
03	Follow the recommended repair action contained in the Last Failure Code field as shown in Table 7–1.
04	Two possible problem sources are indicated: • In the case of a shelf with dual power supplies, one of the power supplies has failed. Follow repair action 07 for the power supply with the Power LED out. • One of the shelf blowers has failed. Follow repair action 06.

Table 7–2 Repair Action Codes (continued)

05	Four possible problem sources are indicated: •Total power supply failure on a shelf. Follow repair action 09. •A device inserted into a shelf that has a broken internal SBB connector. Follow repair action 0A. •A standalone device is connected to the controller with an incorrect cable. Follow repair action 08. •A controller hardware failure. Follow repair action 20.
06	Determine which blower has failed and replace it.
07	Replace power supply.
08	Replace the cable. Refer to the specific device documentation.
09	Determine power failure cause.
0A	Determine which SBB has a failed connector and replace it.
0D	The EMU (Environmental Monitor Unit) has detected an elevated temperature condition. Check the shelf and its components for the cause of the fault.
0E	The EMU (Environmental Monitor Unit) has detected an external air-sense fault. Check components outside of the shelf for the cause of the fault.
0F	An environmental fault previously- detected by the EMU is now fixed. The EIP is used to notify that the repair was successful.
20	Replace the controller module.
22	Replace the indicated cache module.
23	Replace the indicated write cache battery. CAUTION: BATTERY REPLACEMENT MAY CAUSE INJURY.
24	Check for the following invalid write cache configurations: •If the wrong write cache module, replace with the matching module or clear the invalid cache error via the CLI (refer to CLI User Guide). •If the write cache module is missing, re-seat cache if it is actually present, or add the missing cache module or clear the invalid cache error via the CLI (refer to the CLI User Guide). •If in a dual-redundant configuration and one of the write cache modules is missing, match write cache boards with both controllers. •If in a dual-redundant configuration and both caches are not of the same type (i.e., both write cache), replace a cache module to assure both are compatible. •If in a dual-redundant configuration and both write caches are not of the same size, replace a cache module to assure both are compatible.
25	An unrecoverable Memory System failure occurred. Upon restart the controller will generate one or more Memory System Failure Event Sense Data Responses; follow the repair action(s) contained therein.
26	The Master DRAB detected a Cache Timeout condition. The cache regions in effect are identified in the Master DRAB RSR register: bits 8 through 11 identify the CACHEA memory region, bits 12 through 15 identify the CACHEB memory region, bits 20 through 23 the CACHEA DRAB registers region, and bits 24 through 27 identify the CACHEB DRAB registers region. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 36.
27	The Master DRAB detected an Nbus Transfer Error Acknowledge (TEA) condition. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 36.

Table 7–2 Repair Action Codes (continued)

28	A Multiple Bit ECC error was detected by the Master DRAB. The Master DRAB DER register bits 0 through 6 contain the syndrome value. The Master DRAB EAR register combined with Master DRAB ERR bits 0 through 3 (address region) yields the affected memory address. The Master DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 34.
29	A Multiple Bit ECC error was detected by the CACHEA0 or CACHEA1 DRAB. The CACHEAn DRAB DER register bits 0 through 6 contain the syndrome value. The CACHEAn DRAB EAR register combined with the Master DRAB RSR register bits 8 through 11 (CACHEA memory region) yields the affected memory address. The CACHEAn DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 34.
2A	A Multiple Bit ECC error was detected by the CACHEB0 or CACHEB1 DRAB. The CACHEBn DRAB DER register bits 0 through 6 contain the syndrome value. The CACHEBn DRAB EAR register combined with the Master DRAB RSR register bits 12 through 15 (CACHEB memory region) yields the affected memory address. The CACHEBn DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 34.
2B	The Master DRAB detected an Ibus to Nbus Timeout condition. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. If Master DRAB CSR register bits 10 through 12 contains the value 1 and WDR1 register bit 26 is clear or Master DRAB CSR register bits 10 through 12 contains the value 2 and WDR1 register bit 27 is clear or Master DRAB CSR register bits 10 through 12 contains the value 3 and WDR1 register bit 28 is clear or Master DRAB CSR register bits 10 through 12 contains the value 4 and WDR1 register bit 29 is clear or Master DRAB CSR register bits 10 through 12 contains the value 5 and WDR1 register bit 30 is clear or Master DRAB CSR register bits 10 through 12 contains the value 6 and WDR1 register bit 31 is clear, a firmware fault is indicated; follow repair action 01. Otherwise, follow repair action 36.
2C	The Master DRAB detected a Nonexistent Memory Error condition. The Master DRAB EAR register combined with Master DRAB ERR bits 0 through 3 (address region) yields the affected memory address. The Master DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 36.

Table 7–2 Repair Action Codes (continued)

2D	The CACHEA0 or CACHEA1 DRAB detected a Nonexistent Memory Error condition. The CACHEAn DRAB EAR register combined with the Master DRAB RSR register bits 8 through 11 (CACHEA memory region) yields the affected memory address. The CACHEAn DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 36.
2E	The CACHEB0 or CACHEB1 DRAB detected a Nonexistent Memory Error condition. The CACHEBn DRAB EAR register combined with the Master DRAB RSR register bits 12 through 15 (CACHEB memory region) yields the affected memory address. The CACHEBn DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. Follow repair action 36.
2F	The Master DRAB detected an Address Parity Error or a Write Data Parity Error condition. The Master DRAB EAR register combined with Master DRAB ERR bits 0 through 3 (address region) yields the affected memory address. The Master DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. For Write Data Parity Error conditions Bits 0 through 3 of the Master DRAB CSR register identify the byte in error. For Address Parity Error conditions follow repair action 34. For Write Data Parity Error conditions follow repair action 35.
30	The CACHEA0 or CACHEA1 DRAB detected an Address Parity Error or a Write Data Parity Error condition. If the failure occurred during a memory refresh attempt, the CACHEAn DRAB EAR register combined with the Master DRAB RSR register bits 8 through 11 (CACHEA memory region) yields the affected memory address. If the failure occurred during a memory access attempt, the CACHEA0 DRAB EAR register combined with the Master DRAB RSR register bits 8 through 11 (CACHEA memory region) or bits 20 through 23 (CACHEA DRAB register region) yields the affected memory address. Unfortunately, no other information is available to distinguish a memory region access from a DRAB register region access. The CACHEAn DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is set, the failure was reported via the DRAB_INT. For Write Data Parity Error conditions bits 0 through 3 of the CACHEAn DRAB CSR register identify the byte in error. For Address Parity Error conditions follow repair action 34. For Write Data Parity Error conditions follow repair action 35.

Table 7–2 Repair Action Codes (continued)

31	The CACHEB0 or CACHEB1 DRAB detected an Address Parity Error or a Write Data Parity Error condition. If the failure occurred during a memory refresh attempt, the CACHEBn DRAB EAR register combined with the Master DRAB RSR register bits 8 through 11 (CACHEB memory region) yields the affected memory address. If the failure occurred during a memory access attempt, the CACHEB0 DRAB EAR register combined with the Master DRAB RSR register bits 8 through 11 (CACHEB memory region) or bits 20 through 23 (CACHEB DRAB register region) yields the affected memory address. Unfortunately, no other information is available to distinguish a memory region access from a DRAB register region access. The CACHEBn DRAB EDR register contains the error data. If the failure involved a Device Port, the Master DRAB CSR register bits 10 through 12 identify that Device Port. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. For Write Data Parity Error conditions bits 0 through 3 of the CACHEBn DRAB CSR register identify the byte in error. For Address Parity Error conditions follow repair action 34. For Write Data Parity Error conditions follow repair action 35.
32	The Master DRAB detected an Ibus Parity Error condition. The Master DRAB EAR register combined with the Master DRAB ERR bits 4 through 7 (address region) yields the affected memory address. The Master DRAB EDR register contains the error data. If Master DRAB DSR register bit 14 is set, the failure was reported via the NMI. If Master DRAB DSR register bit 14 is clear, the failure was reported via the DRAB_INT. If bits 20 through 23 of the Master DRAB DCSR register contain a nonzero value, a firmware fault is indicated; follow repair action 01. Otherwise, follow repair action 36.
33	This event report contains supplemental information related to a Memory System Failure event report delivered earlier. Use the Instance Code contained in the Memory Address field of this event report to correlate this event report with the other event report.
34	If Bit 31 of the DCSR register of the DRAB that detected the failure is set, a firmware fault is indicated; follow repair action 01. Otherwise, follow repair action 36.
35	If bits 20 through 23 of the WDR1 register contain a nonzero value, a firmware fault is indicated; follow repair action 01. Otherwise, follow repair action 36.
36	Unfortunately, no other information is available to aid in diagnosing the cause of the failure. If the Master DRAB detected the failure, follow repair action 20. If the CACHEAn or CACHEBn DRAB detected the failure, follow repair action 22. If the problem persists, follow repair action 01.
37	The Memory System Failure translator could not determine the failure cause. Follow repair action 01.
40	If the Sense Data FRU field is non-zero, follow repair action 41. Otherwise, replace the appropriate FRU associated with the device's SCSI interface or the entire device.
41	Consult the device's maintenance manual for guidance on replacing the indicated device FRU.
43	Update the configuration data to correct the problem.
44	Replace the SCSI cable for the failing SCSI bus. If the problem persists, replace the controller backplane, drive backplane, or controller module.
45	Interpreting the device supplied Sense Data is beyond the scope of the controller's firmware. See the device's service manual to determine the appropriate repair action, if any.

7.6 Clearing Outdated Messages

Old event or error messages can continue to reappear at the terminal each time you press the Return key. For example, the following message can continue to reappear even after completing the repair of the associated shelf failure.

```
Shelf 1 has bad power supply or fan
shelf 1 fixed
```

To clear out of date messages when they continue to appear, type the following:

HSZ20> **clear_errors cli**

7.7 Avoiding Unwanted Unwritten Cached Data Conditions

Write-back cache is required for RAID functionality. Therefore, you may experience situations that leave unwanted unwritten cache data in the write-back cache module. This section should help remedy some of those situations.

Unwritten cached data can remain in cache for reasons other than a power failure. To avoid some of these situations when write-back cache is enabled, follow these recommendations:

- When write-back caching is enabled on a unit and that unit fails, the cached data for that unit is unwriteable. The data for other write-back cached units on the same controller are still intact and the write-back cache module is still fully functional. However, in order to clear that portion of the write-back cache module's memory that now contains data from the failed unit, you need to enter the CLEAR_ERRORS UNWRITEABLE_DATA command.
- In order to force a cache data flush of an *online* unit, enter the SET *unit* NOWRITEBACK_CACHE command. This forces a flush of any outstanding write data (completes within several minutes).
- If there is inactivity across the bus of the destination device for more than the time set by the CACHE_FLUSH_TIMER= qualifier (or the 10 second default), any outstanding write data is automatically flushed to the inactive devices.
- Write-back cache modules must not be moved from their backplane slots *unless* all unwritten write cached data has been flushed. To determine whether all cache data has been flushed, use the SHOW THIS_CONTROLLER (or SHOW OTHER_CONTROLLER) command to check cache status.
- In the event of a write-back cache module mismatch due to wrong cache module placement, the CLI error INVALID_CACHE is displayed. This error indicates that the controller detects unwritten cached data outstanding for its cache, but that the cache module in the slot is not the correct one. The serial number of the write-back cache module that belongs in the slot is given.
- You may use the CLI command CLEAR_ERRORS INVALID_CACHE to clear errors associate*d with a cache module mismatch. Do not* use this command except in cases of hardware failures or hardware upgrades. *Always* attempt to first find and install the correct cache module, because entering this command will *destroy* the unwritten cached data.

7.8 Using CLI Commands with Write-Back Cache

You can specify whether you want write-back cache enabled (with the WRITEBACK_CACHE qualifier) or disabled (with the NOWRITEBACK_CACHE qualifier) when you initially issue an ADD UNIT command. The NOWRITEBACK_CACHE qualifier (write-through) is the default.

After a unit is added, if you want to disable or enable your write-back cache, enter the CLI SET *unit-number* command and specify either the WRITEBACK_CACHE qualifier or the NOWRITEBACK_CACHE qualifier.

A write-back cache module is required for RAID operations.

Write-back caching, RAID 3/5, and mirroring are separately licensed features. You can use these licensed features on a trial basis, but if you do not buy the required licenses an hourly message is sent to your console terminal and to the host error log. Once you buy the required licenses, you are given license keys to be used with the **firmware licensing system (FLS)** utility. After the keys are entered, the messages stop. See Chapter 4 for further information.

When the power for the write-back cache module batteries is too low, a console message is displayed. To check the status of the batteries on the write-back cache module, enter the SHOW THIS_CONTROLLER or SHOW OTHER_CONTROLLER command at the CLI prompt. The battery status of the cache module is displayed as good, low, or bad.

A

RAID Array 310 CLI Commands

This chapters describes each of the command line interface (CLI) commands including syntax and examples of how to use them. In addition, this chapter describes the error and warning messages that you may encounter while using the CLI.

A.1 RAID Array 310 CLI Commands

The following section describes the user commands available in the CLI, including syntax, parameters, qualifiers, and examples of how to use each command. The user commands include the following:

Add Disk
Add Mirrorset
Add RAIDset
Add Spareset
Add Stripeset
Add Unit
Clear_Errors CLI
Clear_Errors Invalid Cache
Clear_Errors Lost_Data
Clear_Errors Unknown
Clear_Errors Unwriteable_Data
Create Partition
Delete *container-name*
Delete Failedset
Delete Spareset
Delete *Unit-Number*
Destroy Partition
Directory
Exit
Help
Initialize
Locate
Mirror
Reduce
Rename
Restart This_Controller
Retry_Errors Unwriteable_Data
Run
Selftest This_Controller
Set *disk-container-name*
Set Failedset
Set *mirrorset-container-name*
Set *RAIDset-container-name*
Set This_Controller
Set *unit-number*
Show Devices
Show Disks
Show *disk-container-name*
Show Failedset
Show Mirrorsets
Show *mirrorset-container-name*
Show RAIDsets
Show *raidset-container-name*
Show Spareset
Show Storagesets
Show Stripesets
Show *stripeset-container-name*
Show This_Controller
Show Units
Show Unit Number
Shutdown This_Controller
Unmirror *disk-device-name*

ADD DISK

Adds a disk drive to the list of known disk drives and names the drive. This command must be used when a new SCSI–2 disk drive is to be added to the configuration.

Syntax

```
ADD DISK CONTAINER-NAME SCSI-LOCATION
```

Parameters

container-name

Specifies the name that is used to refer to this disk drive. This name is referred to when creating units and stripesets. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up of letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

SCSI-location

The location of the disk drive to be added in the form PTL where **P** designates the port (1–6 or 1–2, depending on the controller model), **T** designates the target ID of the device, 0–6, and **L** designates the LUN of the device (must be 0).

When entering the PTL, at least one space must separate the port, target, and LUN numbers.

Qualifiers

TRANSPORTABLE/NOTRANSPORTABLE (Default)

In normal operations, the controller makes a small portion of the disk inaccessible to the host and uses this area to store metadata, which improves data reliability, error detection, and recovery. This vast improvement comes at the expense of transportability.

If NOTRANSPORTABLE is specified and there is no valid metadata on the unit, the unit must be initialized.

NOTE

Digital recommends that you avoid specifying TRANSPORTABLE unless transportability of disk drive or media is imperative and there is no other way to accomplish moving the data.

Examples

HSZ20> **ADD DISK RZ26_100 1 0 0**
Adds a non transportable disk to port 1, target 0, LUN 0 and names it RZ26_100.

HSZ20> **ADD DISK DISK0 2 3 0 NOTRANSPORTABLE**
Adds a non transportable disk to port 2, target 3, LUN 0 and names it DISK0.

HSZ20> **ADD DISK TDISK0 3 2 0 TRANSPORTABLE**
Adds a transportable disk to port 3, target 2, LUN 0 and names it TDISK0.

ADD MIRRORSET

Binds a set of physical devices to a mirrorset specified by a container name. Adds a mirrorset to the list of known mirrorsets and names the mirrorset. The number of members is set to the number of devices specified in the command. If the physical devices have never been initialized as a mirrorset, then a CLI INITIALIZE command must be issued for the mirrorset prior to binding the mirrorset to a higher level storage container.

If the CACHE_POLICY for a controller is set to B, the controller allows the ADD MIRRORSET command, even if the write-back cache battery is low. If the CACHE_POLICY is set to A and the batteries are low, the command will fail. Policy A requires that the cache batteries be fully charged before you can add mirrorsets to your configuration.

Following are some guidelines for creating mirrorsets:
A mirrorset can have a maximum of 6 members.
There can be a maximum of 20 mirrorsets and/or RAIDsets..
There can be a maximum of 30 storagesets (that is, RAIDsets, mirrorsets, and/or stripesets.)

Syntax

ADD MIRRORSET *CONTAINER-NAME DISK-DEVICE-NAME1 [DISK-DEVICE-NAMEn*]

Parameters

container-name
Specifies the name that is used to refer to this mirrorset container. This name is referred to when creating mirrorsets. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up of letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

disk-device-name1 disk-device-nameN
The disk drives that make up this mirrorset. A mirrorset is made up of from 1 to 6 disk drives.

Qualifiers

COPY=copy_speed
The COPY qualifier allows you to specify the speed at which mirrorset copies are performed. You may specify either NORMAL or FAST.
NORMAL uses relatively few controller resources to perform the copy, and has little impact on controller performance.
FAST uses more controller resources, which reduces the time it takes to complete the copy, but also reduces overall controller performance.

POLICY=BEST_FIT\POLICY=BEST_PERFORMANCE (Default)\NOPOLICY

Examples The POLICY qualifier specifies the replacement policy to be used when a mirrorset member within the mirrorset fails.

BEST_FIT gives highest priority to finding a replacement device within the spareset that most closely matches the sizes of the remaining members of the mirrorset. If more than one device in the spareset is the correct size, the device that gives the best performance is selected.

BEST_PERFORMANCE (default) gives highest priority to finding a replacement device within the spareset that results in the best performance of the mirrorset (the device should be on a different port). If more than one device in the spareset has the best performance, the device that most closely matches the size of the remaining members of the mirrorset is selected.

NOPOLICY retries a failing device from the mirrorset without selecting a replacement. This causes the mirrorset to run with less than the nominal number of members until a BEST_FIT or BEST_PERFORMANCE policy is selected, or a member is manually replaced in the mirrorset.

READ_SOURCE=read-source

The READ_SOURCE qualifier allows you to control the read algorithm for the specified mirrorset. The following choices are allowed for read-source:

ROUND_ROBIN—Each NORMAL mirrorset member is the target of a read in sequential membership order. No preference is given to any NORMAL member.

LEAST_BUSY—The NORMAL mirrorset member with the least busy work queue is the target of the read. This is the default READ_SOURCE.

DEVICE-CONTAINER-NAME—All reads are done on device-container-name. If *device-container-name* fails out of the mirrorset, the READ_SOURCE algorithm reverts to LEAST_BUSY.

Examples

`HSZ20>` **`ADD MIRRORSET MIRR1 DISK100 DISK210 DISK320`**
Adds DISK100, DISK210, and DISK320 as a mirrorset with the name MIRR1.

ADD RAIDSET

Creates a RAIDset from a number of containers. Adds a RAIDset to the list of known RAIDsets and names the RAIDset. This command must be used when a new RAIDset is to be added to the configuration.

If the CACHE_POLICY for a controller is set to B, the controller allows the ADD RAIDSET command, even if the write-back cache battery is low. If the CACHE_POLICY is set to A and the batteries are low, the command will fail. Policy A requires that the cache batteries be fully charged before you can add RAIDsets to your configuration.

Following are some guidelines for creating RAIDsets:

There can be a maximum of 20 mirrorsets and/or RAIDsets..

There can be a maximum of 30 storagesets (that is, RAIDsets, mirrorsets, and/or stripesets.)

Syntax

ADD RAIDSET CONTAINER-*NAME CONTAINER-NAME1 CONTAINER-NAME2 [CONTAINER-NAMEn]*

Parameters

container-name
Specifies the name that is used to refer to this RAIDset. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up of letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

container-name1 container-name2 container-nameN
The containers that will make up this RAIDset. A RAIDset may be made up of from 3 to 14 containers.

Qualifiers

POLICY=BEST_FIT\POLICY=BEST_PERFORMANCE (Default)\NOPOLICY
Specifies the replacement policy to use when a member within the RAIDset fails.
BEST_FIT gives highest priority to finding a replacement device within the spareset that most closely matches the sizes of the remaining members of the RAIDset. If more than one device in the spareset is the correct size, the device that gives the best performance is selected.
BEST_PERFORMANCE (default) gives highest priority to finding a replacement device within the spareset that results in the best performance of the RAIDset. If more than one device in the spareset has the best performance, the device that most closely matches the size of the remaining members of the RAIDset is selected.
NOPOLICY retires a failing device from the RAIDset without selecting a replacement. This causes the RAIDset to run in a reduced state until a BEST_FIT or BEST_PERFORMANCE policy is selected, or a member is manually replaced in the RAIDset (see SET *RAIDset-container-name*).

RECONSTRUCT=NORMAL (Default)

Specifies the speed at which a RAIDset will be reconstructed when a new member is added to the RAIDset or immediately after the RAIDset is initialized.

RECONSTRUCT=NORMAL (default) balances overall performance of the controller against the demand of reconstructing the RAIDset.

RECONSTRUCT=FAST compromises overall performance of the controller to speed reconstructing the RAIDset.

REDUCED\NOREDUCED (Default)

REDUCED specifies that the RAIDset being added is already missing one member. Use the REDUCED keyword when moving an already reduced RAIDset from one controller to another. NOREDUCED (default) identifies that all RAIDset members that make up the RAIDset are being specified.

Examples

`HSZ20>` **`ADD RAIDSET RAID9 DISK0 DISK1 DISK2 DISK3`**
Creates a RAIDset with four disks (DISK0, DISK1, DISK2, and DISK3). The replacement policy is BEST_PERFORMANCE.

`HSZ20>` **`ADD RAIDSET RAID9 DISK0 DISK1 DISK2 DISK3 POLICY=BEST_FIT`**
Creates a RAIDset with four disks (DISK0, DISK1, DISK2, and DISK3). The replacement policy is BEST_FIT, as specified.

`HSZ20>` **`ADD RAIDSET RAID9 DISK0 DISK1 DISK2 DISK3 NOPOLICY`**
Creates a RAIDset with four disks (DISK0, DISK1, DISK2, and DISK3). If a member within the RAIDset fails, a replacement will *not* be selected.

`HSZ20>` **`ADD RAIDSET RAID9 DISK0 DISK1 DISK3 REDUCED`**
Creates a four member RAIDset with a raidset that was already reduced.

ADD SPARESET

Adds a disk drive to the spareset. The spareset is a pool of drives available to the controller to replace failing members of RAIDsets and mirrorsets. The ADD SPARESET command adds a disk drive to the spareset and initializes the metadata on the drive.

There can be a maximum of 30 storagesets (that is, RAIDsets, mirrorsets, and/or stripesets.)

Syntax

`ADD SPARESET` ***`DISK-CONTAINER-NAME`***

Parameters

disk-container-name
The disk drive container name to add to the spareset. Only one disk may be added to the spareset at a time.

Examples

`HSZ20>` **`ADD SPARESET DISK220`**
Adds one disk to the spareset.

ADD STRIPESET

Creates a stripeset from a number of containers. Adds a stripeset to the list of known stripesets and names the stripeset. This command must be used when a new stripeset is added to the configuration.

There can be a maximum of 30 storagesets (that is, RAIDsets, mirrorsets, and/or stripesets.)

Syntax

ADD STRIPESET ***CONTAINER-NAME CONTAINER-NAME1 CONTAINER-NAME2 [CONTAINER-NAMEn***

Parameters

container-name
Specifies the name that is used to refer to this stripeset. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up of letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

container-name1 container-name2 container-nameN
The containers that will make up this stripeset. A stripeset may be made up of from 2 to 14 containers.

Examples

`HSZ20>` **`ADD STRIPESET STRIPE0 DISK100 DISK110 DISK220 DISK340`**
Creates a STRIPESET with four disks (DISK100, DISK110, DISK220, and DISK340).

`HSZ20>` **`ADD STRIPESET STRIPE1 MR1 MR2 MR3`**
Creates a STRIPESET with three members, each of which is a mirrorset.

ADD UNIT

Adds a logical unit to the controller. The ADD UNIT command adds a logical unit for the host to access. All requests by the host to the logical unit number are mapped as requests to the container specified in the ADD UNIT command.

For disk devices (and stripesets, RAIDsets, and mirrorsets built from disk devices), the metadata on the container must be initialized before a unit may be created from it. If the container's metadata cannot be found, or is incorrect, an error is displayed and the unit is not created.

The maximum number of physical device members in a unit is 32.

Syntax

ADD UNIT *UNIT-NUMBER CONTAINER-NAME*

Parameters

unit-number
The unit number determines both the target (0–7) and the LUN (0–7) from which the device is made available. The hundreds place of the unit number is the target and the ones place is the LUN. The tens place is not currently used. For example, D401 would be target 4, LUN 1; D100 would be target 1, LUN 0, and D5 would be target 0, LUN 5.

NOTE

The only target numbers specified in the unit number *must* have been previously specified in the SET THIS_CONTROLLER ID=(*n*1, *n*2, ...) command. You can not specify a target number that has not been previously specified by the SET THIS_CONTROLLER ID=(*n*1, *n*2, ...) command.

container-name
The name of the container that is used to create the unit.

Qualifiers for a Unit Created from a TRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

READ_CACHE (Default)\NOREAD_CACHE
Enables and disables the controller's read cache on this unit.

RUN (Default)\NORUN
Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

Qualifiers for a Unit Created from a NOTRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

READ_CACHE (Default)\NOREAD_CACHE
Enables and disables the controller's read cache on this unit.

RUN (Default)\NORUN
Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

WRITEBACK_CACHE\NOWRITEBACK_CACHE (Default)
Enables and disables the controller's write-back cache on this unit.

NOTE

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching. When initially added, NOWRITEBACK_CACHE is the default.

Qualifiers for a Unit Created from a RAIDset

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

RUN (Default)\NORUN
Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

NOTE

Writes may still be performed to a write-protected RAIDset to satisfy a reconstruct pass or to reconstruct a newly replaced member. However, write protect will disable the writing of any new data.

WRITEBACK_CACHE\NOWRITEBACK_CACHE (Default)
Enables and disables the controller's write-back cache on this unit.

NOTE

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Qualifiers for a Unit Created from a Stripeset

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

READ_CACHE (Default)\NOREAD_CACHE
Enables and disables the controller's read cache on this unit.

RUN (Default)\NORUN
Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

WRITEBACK_CACHE\NOWRITEBACK_CACHE (Default)
Enables and disables the controller's write-back cache on this unit.

NOTE

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Examples

```
HSZ20> ADD UNIT D0 DISK0
```
Creates disk unit number 0 from container DISK0.

```
HSZ20> ADD UNIT D107 RAID9 WRITE_PROTECT
```
Creates disk unit number 107 from container RAID9 and write protects it.

CLEAR_ERRORS CLI

Stops displaying errors at the CLI prompt. Errors detected by controller firmware are displayed before the CLI prompt. These errors are displayed even after the error condition is rectified, until the controller is restarted or the CLEAR_ERRORS CLI command is issued.

NOTE

This command does not clear the error conditions, it only clears displaying the errors at the CLI prompt.

Syntax

`CLEAR_ERRORS CLI`

Examples

```
HSZ20> ALL NVPM COMPONENTS INITIALIZED TO THEIR DEFAULT
SETTINGS.
HSZ20> CLEAR_ERRORS CLI
HSZ20>
```

Clears the message “All NVPM components initialized to their default settings.” that was displayed at the CLI prompt.

CLEAR_ERRORS INVALID_CACHE

Clears an INVALID_CACHE error condition and makes a mismatched cache module usable by the specified controller.

NOTE

The INVALID_CACHE portion of the command must be completely spelled out, not abbreviated.

The controller module, cache module, and subsystem devices all contain configuration information, called metadata, used to keep their activity synchronized.

The firmware reports an INVALID_CACHE error on the affected controller in situations in which the metadata in a cache module containing unwritten data has become incompatible with that in the controller module. This mismatch can cause the loss of the unwritten cache data. Following are several examples in which a metadata mismatch can occur:

Mismatches Caused by Changes in Controller Metadata

A mismatch caused by changes in controller metadata can occur in the following situations:

- You replace the controller in your subsystem with a factory replacement or with one from a "foreign" subsystem.
- The controller's nonvolatile memory or memory battery fails.

The controller reports an INVALID_CACHE error to alert you to a cache/controller module mismatch and the possibility of losing your unwritten cache data.

Any unwritten cache data, user data resulting from normal subsystem activity before the controller's metadata changed, is valuable.

To make sure that the controller can recover the data, clear the error with the CLEAR_ERRORS INVALID_CACHE_NODEDTROY_UNFLUSHED_DATA command. This command clears the error without deleting the unwritten cache data, and synchronizes the controller metadata with that of the cache module. The controller can then recover the unwritten data by writing it to the appropriate devices.

Mismatches Caused by Changes in Cache Metadata

A mismatch caused by changes in cache metadata can occur in the following situations:

- You replace the cache module with a factory replacement or with one from a "foreign" subsystem.
- You initialize the cache module using the cache backup battery disable jumper during a module replacement action.
- You turn on the power to a new subsystem for the first time.

The controller reports an INVALID_CACHE error to alert you to a cache/controller module mismatch.

Any unwritten cache data is not valuable user data from your subsystem and does not need to be recovered.

To clear the error, use the CLEAR_ERRORS INVALID_CACHE_DESTROY_UNFLUSHED_DATA command. This clears the error, clears the cache of any unwritten data, and synchronizes the controller metadata with that of the cache module.

> **NOTE**
>
> If you clear the error with the CLEAR_ERRORS INVALID_CACHE_NODESTROY_UNFLUSHED_DATA command, any residual data in the cache remains. This does not cause an error condition, but does consume valuable cache memory.

Mismatches Caused by Removal of the Cache Module

The controller may report an invalid cache error if you remove a cache module containing unwritten cache data from your controller. The controller reports an INVALID_CACHE error to alert you to potential of losing unwritten cache data.

Any unwritten cache data is valuable user data resulting from normal subsystem activity before you removed the cache module. The unwritten data is physically unavailable to the subsystem if you do not reinstall the original cache module. If you reinstall the original cache module, the controller can recover the unwritten data by writing it to the appropriate devices.

To clear the error, use the CLEAR_ERRORS INVALID_CACHE_NODESTROY_UNFLUSHED_DATA command. This clears the error, and causes the controller to operate in a degraded mode (write-back caching is disabled and RAIDsets and mirrorsets are inoperative). The controller uses a small portion of its RAM memory for read caching only.

If you do not clear an invalid cache error by issuing a CLEAR_ERRORS INVALID_CACHE command, the controller is prevented from completely booting. The controller boots to a point where a limited CLI command set is available to the user. It remains unavailable to the host until the user clears the error.

> **NOTE**
>
> Issuing the CLEAR_ERRORS INVALID_CACHE on THIS controller forces a restart on the OTHER controller, in dual-redundant configurations.

Syntax

`CLEAR_ERRORS INVALID_CACHE` ***`CONTROLLER`***

Parameters

controller
Specifies in which controller the INVALID_CACHE condition is cleared. THIS_CONTROLLER must be specified.

nodestroy_unflushed_data
Specifies to preserve unwritten cache data and flush it to the appropriate disk devices. Causes the controller to synchronize its metadata with that of the cache module and preserves any unwritten cache data for eventual flushing to the storage array.

destroy_unflushed_data
Specifies not to preserve unwritten cache data or to flush it to the appropriate disk devices. Causes the controller to synchronize its metadata with that of the cache module and delete all unwritten cache data.

> **NOTE**
>
> There is no default for the data retention (destroy/nodestroy) parameters for this command. Also, you must specify these parameters completely and cannot abbreviate them.

Examples

HSZ20> **CLEAR_ERRORS INVALID_CACHE THIS_CONTROLLER**
Clears the INVALID_CACHE error.

CLEAR_ERRORS LOST_DATA

Clears the LOST_DATA error on a unit.

> **NOTE**
>
> The LOST_DATA portion of the command must be completely spelled out, not abbreviated. Once the command is issued, it may take up to *5 minutes* to clear the lost data error.

If customer data has been lost because of the removal or failure of the write-back cache module, a LOST_DATA error is reported on the units affected. A unit that is reported as having lost data is automatically write protected until the error condition is cleared. The CLEAR_ERRORS LOST_DATA command clears the LOST_DATA error and write-protect condition, and updates the media metadata on the specified unit.

Note that issuance of the CLEAR_ERRORS LOST_DATA command does not itself cause data loss. It is the removal or failure of the cache module that may cause a data loss. The command only affects the error condition and cannot recover customer data.

Syntax

`CLEAR_ERRORS LOST_DATA` ***`UNIT-NUMBER`***

Parameters

unit-number
Specifies the logical unit number (D0–D7, D100-D107, and so forth) on which the lost data error is to be cleared. The *unit-number* is the name given the unit when it was created using the ADD UNIT command.

Examples

`HSZ20>` **`CLEAR_ERRORS LOST_DATA D103`**
Clears the LOST_DATA error on disk unit D103.

CLEAR_ERRORS UNKNOWN

Clears the UNKNOWN error from a device.

> **NOTE**
>
> "UNKNOWN" must be completely spelled out, not abbreviated.

If a device has a failure such that the controller marks the device as UNKNOWN, the device is never automatically checked again to see if it has been repaired or if the failure condition was rectified. When you rectify a condition that caused a device to be marked UNKNOWN, this command must be issued for the controller to recognize the device.

Syntax

`CLEAR_ERRORS UNKNOWN` ***`DEVICE-NAME`***

Parameters

device-name
Specifies the device name of the device with the UNKNOWN error.

Examples

`HSZ20>` **`CLEAR_ERRORS UNKNOWN DISK300`**
Causes the controller to recognize DISK300, a previously UNKNOWN device.

CLEAR_ERRORS UNWRITEABLE_DATA

Clears the unwriteable data error on a unit.

CAUTION

This command causes loss of customer data.

NOTE

Because this command causes loss of customer data, "UNWRITEABLE_DATA" must be completely spelled out, not abbreviated.

If a container fails in a way that customer data in the write-back cache cannot be written to the container, the unwriteable data error is reported. CLEAR_ERRORS UNWRITEABLE_DATA clears the unwriteable data error, however, *all customer data that has not been written to disk is lost*.

For this reason, use great caution when considering using this command.

Syntax

CLEAR_ERRORS UNWRITEABLE_DATA *UNIT-NUMBER*

Parameters

unit-number
Specifies the logical unit number (D0–D7, D100-D107, and so forth) that will have the unwriteable data error cleared. The *unit-number* is the name given the unit when it was created using the ADD UNIT command.

Examples

HSZ20> **CLEAR_ERRORS UNWRITEABLE_DATA D103**

CREATE-PARTITION

Marks part of a disk or storageset to be used as a separately addressable unit. You can divide any disk or storageset into up to eight partitions, each of which can be separately presented to the host.

> **NOTE**
>
> Once you have partitioned a container, you cannot reunify the container without reinitializing the container

Syntax

CREATE_PARTITION ***container-name*** **SIZE=n**

Parameters

container-name

The name of the disk or storageset on which you want to create a partition. This is the name given the disk or storageset when it was created using the ADD command (ADD DISK, ADD STRIPESET, and so forth). You can partition any disk, stripeset, mirrorset, striped ,mirrorset, or RAIDset. You must initialize the container before creating partitions.

SIZE=*percent*
SIZE-LARGEST

Specified the size of the partitions to be created as a percentage of the total size of the disk or storageset.

Specify a percentage of the total container capacity to set the partition to a particular size, or to evenly divide the disk or storageset. The resulting partition will be slightly smaller than the size you specify, to accommodate controller information and so that each partition ends with a complete stripe.

Specify LARGEST to have the controller create the largest partition possible from unused space on the disk or storageset. You must also use the LARGEST setting when creating the last partition on a container, since the space will not be equal to an exact percentage value.

Examples

To create RAIDset R9 and divide it into four parts:

```
HSZ20> ADD DISK DISK100 1 0 0
HSZ20> ADD DISK DISK210 2 1 0
HSZ20> ADD DISK DISK320 3 2 0
HSZ20> ADD RAIDSET9 DISK100 DISK210 DISK320
HSZ20> INITIALIZE RAID9
HSZ20> CREATE PARTITIONS RAID9 SIZE=25
HSZ20> CREATE PARTITIONS RAID9 SIZE=25
HSZ20> CREATE PARTITIONS RAID9 SIZE=25
HSZ20> CREATE PARTITIONS RAID9 SIZE=LARGEST
HSZ20> ADD UNIT D101 RAID9 PARTITION=1
HSZ20> ADD UNIT D101 RAID9 PARTITION=2
HSZ20> ADD UNIT D101 RAID9 PARTITION=3
HSZ20> ADD UNIT D101 RAID9 PARTITION=4
```

DELETE CONTAINER-NAME

Deletes a container from the list of known containers. Checks to see if the container is used by any other containers or a unit. If the container is in use, an error is displayed and the container is not deleted.

If the container is not in use, it is deleted.

NOTE

The spareset and failedset containers cannot be deleted. See DELETE SPARESET and DELETE FAILEDSET commands.

Syntax

DELETE *CONTAINER-NAME*

Parameters

container-name
Specifies the name that identifies the container. This is the name given the container when it was created using the ADD command (ADD DEVICE, ADD STRIPESET, and so forth).

Examples

HSZ20> **DELETE DISK0**
Deletes DISK0 from the list of known containers.

HSZ20> **DELETE STRIPE0**
Deletes STRIPE0 from the list of known containers.

HSZ20> **DELETE RAID9**
Deletes RAID9 from the list of known containers.

DELETE FAILEDSET

Delete a disk drive from the failedset. The FAILEDSET a group of drives that were removed from RAIDsets and mirrorsets, either because they failed or were manually removed via the SET command, or were physically removed from the system. Drives in the failedset should be considered defective and should be tested, then repaired or replaced. The DELETE FAILEDSET command removes drives from the failedset, typically before you remove them physically from the shelf for testing, repair, or replacement.

Syntax

`DELETE FAILEDSET` ***`DISK-CONTAINER-NAME1 [DISK-CONTAINER-NAME n]`***

Parameters

disk-container-name1 disk-container-nameN
The disk drive container names to delete from the failedset. Any number of disks may be deleted from the failedset using only one command.

Examples

`HSZ20>` **`DELETE FAILEDSET DISK220`**
Deletes one disk from the failedset.

`HSZ20>` **`DELETE FAILEDSET DISK100 DISK210 DISK220 DISK330 DISK400`**
Deletes five disks from the failedset.

DELETE SPARESET

Delete a disk drive from the spareset. The **SPARESET** is a pool of drives available to the controller to replace failing members of raidsets and mirrorsets. The DELETE SPARESET command removes disk drives from the spareset.

Syntax

`DELETE SPARESET` ***`DISK-CONTAINER-NAME0 [DISK-CONTAINER-NAMEN]`***

Parameters

disk-container-name0 disk-container-nameN
The disk drive container names to delete from the spareset. Any number of disks may be deleted from the spareset using only one command.

Examples

`HSZ20>` **`DELETE SPARESET DISK230`**
Deletes one disk from the spareset.

`HSZ20>` **`DELETE SPARESET DISK110 DISK210 DISK240 DISK320 DISK400`**
Deletes five disks from the spareset.

DELETE *UNIT-NUMBER*

Deletes a unit from the list of known units. The DELETE command flushes any user data from the write-back cache to the disk and deletes the logical unit. If any errors occur when trying to flush the user data, the logical unit is not deleted.

In order to delete a unit that has cache errors, you must clear all cache errors associated with the unit via a CLEAR_ERRORS command.

Syntax

DELETE *UNIT-NUMBER*

Parameters

unit-number
Specifies the logical unit number (D0–D7, D100-D107, and so forth) that is to be deleted. The *unit-number* is the name given the unit when it was created using the ADD UNIT command.

Examples

HSZ20> **DELETE D102**
Deletes disk unit number 102 from the list of known units.

DESTROY PARTITION

Marks a used partition as free, and consolidates it with any adjacent free partitions. User data on the partition is lost.

You cannot destroy a partition that has a unit number assigned; you must first use the DELETE *unit-number* command to delete the unit.

Syntax

DESTROY_PARTITION ***container-name*** **PARTITION=*****partition-number***

Parameters

container-name

The name of the disk or storageset that contains the partition that you want to destroy. This is the name given the disk or storageset when it was created using the ADD command (ADD DISK, ADD STRIPESET, and so forth).

partition-number

The number of the partition that you want to destroy. You can use the SHOW command to find partition numbers.

CAUTION

Be sure to use the SHOW command to identify the correct partition before using the DESTROY_PARTITON command

Examples

```
HSZ20> DELETE D102
HSZ20> DESTROY_PARTITION RAID9 PARTITION=2
```

Deletes the unit for partition 2 on RAIDset RAID9 and destroy the partition:

DIRECTORY

Lists the diagnostics and utilities available on THIS_CONTROLLER. The DIRECTORY command lists the various diagnostics and utilities that are available on THIS_CONTROLLER. A directory of diagnostics and utilities available on this controller is displayed.

Syntax

DIRECTORY

Examples

```
HSZ20> DIRECTORY
CLCP   V27Z   D
CONFIG V27Z   D
C_SWAP V27Z   D
CFMENU V27Z   D
CHVSN  V27Z   D
CLONE  V27Z   D
CRASH  V27Z   D
DILX   V27Z   D
DIRECT V27Z   D
FLS    V27Z   D
FMU    V27Z   D
VTDPY  V267Z   D
```

Displays directory listing.

EXIT

Exits the CLI and breaks the virtual terminal connection. When entering the EXIT command from a host using a virtual terminal connection, the connection is broken and control is returned to the host. If entered from a maintenance terminal, the EXIT command restarts the CLI, displaying the copyright notice, the controller type, and the last fail packet.

Syntax

EXIT

Examples

```
HSZ20> EXIT CONTROL RETURNED TO HOST $
```

An EXIT command issued on a terminal that was connected to the CLI via a virtual terminal connection.

HELP

Displays an overview for getting help. The HELP command displays a brief description for using the question mark “?”to obtain help on any command or CLI function.

Syntax

HELP

Examples

```
HSZ20> HELP
HELP MAY BE REQUESTED BY TYPING A QUESTION MARK (?) AT THE CLI
PROMPT. THIS WILL PRINT A LIST OF ALL AVAILABLE COMMANDS

FOR FURTHER INFORMATION YOU MAY ENTER A PARTIAL COMMAND AND TYPE
A SPACE FOLLOWED BY A "?" TO PRINT A LIST OF ALL AVAILABLE
OPTIONS AT THAT POINT IN THE COMMAND. FOR EXAMPLE:
            SET THIS_CONTROLLER ?
PRINTS A LIST OF ALL LEGAL SET THIS_CONTROLLER COMMANDS
```

Displaying help using the HELP command.

```
HSZ20> SET ?
YOUR OPTIONS ARE:
    FAILOVER
    OTHER_CONTROLLER
    NOFAILOVER
    THIS_CONTROLLER
    UNIT NUMBER OR CONTAINER NAME
```

Getting help on the SET command, using the“?”facility.

INITIALIZE

Initializes the metadata on the container specified.

The INITIALIZE command initializes a container so a logical unit may be created from it. During initialization, a small amount of disk space is used for controller metadata and is made inaccessible to the host.

If a single-disk container was set as TRANSPORTABLE, any metadata is destroyed on the device and the full device is accessible to the host.

CAUTION

The INITIALIZE command destroys all customer data on the container.

NOTE

It may take up to 3 minutes to initialize a RAIDset, stripeset, or mirrorset.

The INITIALIZE command is required when:

- A unit is going to be created from a newly installed disk
- A unit is going to be created from a newly created storageset, (RAIDset, stripeset, or mirrorset)
- The INITIALIZE command is *not* required when:
- A unit has been deleted, and a new unit is going to be created from the same container
- A storageset that was initialized in the past is deleted, then added again using the same members that were in the original storageset

Syntax

```
INITIALIZE CONTAINER-NAME
```

Parameters

container-name
Specifies the container name to initialize.

Qualifiers

CHUNKSIZE=n\CHUNKSIZE=DEFAULT (Default)
Specifies the chunksize to be used for RAIDsets and stripesets. You may specify a chunksize in blocks (CHUNKSIZE=*n*), or you can use the defaults. (CHUNKSIZE=DEFAULT).
Table A–1 shows the two default chunksizes.

Table A–1 Default Chunksizes

Controller Model	Default Chunksize
All HSZ20 controller models HSZ20 controllers with 16 MB cache SIMMs	256
HSZ20 controllers with 4 MB cache SIMMs	64

CAUTION

The CLI does not limit the chunksize you can enter. Do not enter a chunksize of over 64 for an SC–4200 controller with a 4 MB cache. A controller hang may result.

Note that the chunksize for SC–4200 controllers using 4 MB cache SIMMs is a *maximum* of 64. This limitation does not apply to controllers using 16 MB cache SIMMS.

CAUTION

Moving a container with a chunksize larger than 64 to an SC4200 controller can cause data corruption on the container.

Do not attempt to create a container with a larger chunksize, and do not bring containers with a larger chunksize from other environments.

The CHUNKSIZE qualifier does not apply to mirrorsets.

Digital recommends that the chunksize for RAIDsets not be set larger than $2048 \div (n–1)$ blocks where n is the number of RAIDset members. Setting larger chunksizes may result in degraded performance and conditions requiring controller reinitialization.

For RAID 3 applications, an approximation of optimum chunksize is $Requestsize \div (n–1)$ where n is the number of RAIDset members.

For controllers with cache sizes greater than 16MB, you can also set the optimum chunksize based on the number of members in the storageset. If you controller has 16MB of cache memory and the number of members in the storageset is less than or equal to nine, use a chunksize of 256 blocks. If you controller has less than 16MB of memory and the number of members in the sotrageset is greater than nine, use a chunksize of 128.

The minimum chunksize for stripesets is 16 blocks (8 KB). The maximum chunksize is $2^{31}–1$ blocks, but because this is larger than any supported disk, it is not a practical limitation.

DESTROY(Default)\NODESTROY

This qualifier prevents the user data and forced error metadata from being destroyed during the initialize. This allows the data on the container to be reused for a disk, stripeset, or mirrorset unit. (The NODESTROY qualifier is ignored for RAIDsets.) NODESTROY is only used when creating a unit out of devices that have been reduced from mirrorsets.

SAVE_CONFIGURATION(Default)\NOSAVE_CONFIGURATION
This qualifier modifies the initialization process to place additional metadata on the device. The additional metadata is used for saving the current subsystem configuration and any changes as they are made. This qualifier is only valid for disk devices. NOSAVE_CONFIGURAITON is the default.

Examples

`HSZ20>` **`INITIALIZE DISK0`**
Initializes container DISK0. If NOTRANSPORTABLE was specified (or allowed to default), metadata is written on the disk.

`HSZ20>` **`INITIALIZE STRIPE0 CHUNKSIZE=20`**
Initializes container STRIPE0 and writes metadata on it. The default chunksize for stripesets is 256 blocks.

`HSZ20>` **`INITIALIZE RAID9 CHUNKSIZE=20`**
Initializes container RAID9 with a chunksize of 20 and writes metadata on it. The default chunksize for RAIDsets is 256 blocks.

`HSZ20>` **`INITIALIZE MIRROR1 NODESTROY`**
Initializes container MIRROR1 and does not write over the forced error metadata.

`HSZ20>` **`INITIALIZE DISK310 SAVE_CONFIGURATION`**
Initializes container DISK310 with subsystem configuration information stored on it.

LOCATE

Locates units, storagesets, and devices. The LOCATE command illuminates the amber device fault LEDs (the innermost LED on the front of an SBB) of the containers specified. The LOCATE command also can be used as a lamp test.

Syntax

`LOCATE`

Qualifiers

ALL
The LOCATE ALL command turns on the amber device fault LEDs of all configured devices. This qualifier also can be used as a lamp test. See LOCATE CANCEL to turn off the LEDs. An error is displayed if no devices have been configured.

CANCEL
The LOCATE CANCEL command turns off all amber device fault LEDs on all configured devices. An error is displayed if no devices have been configured.

DISKS
The LOCATE DISKS command turns on the amber device fault LEDs of all configured disks. See LOCATE CANCEL to turn off the LEDs. An error is displayed if no disks have been configured.

UNITS
The LOCATE UNITS command turns on the amber device fault LEDs of all devices used by units. This command is useful to determine which devices are not currently configured into logical units. See LOCATE CANCEL to turn off device the LEDs. An error is displayed if no units have been configured.

PTL SCSI-location
The LOCATE PTL *SCSI-location* command turns on the amber device fault LEDs at the given SCSI location. *SCSI-location* is specified in the form PTL where **P** designates the port (1–6 or 1–2, depending on the controller model), **T** designates the target ID of the device, 0–6, and **L** designates the LUN of the device (0–7).

When entering the PTL, at least one space must separate the port, target, and LUN numbers. See LOCATE CANCEL to turn off the LEDs.

An error is displayed if the port, target, or LUN is invalid, or if no device is configured at that location.

device or storageset name or unit number (entity)
The LOCATE *entity* command turns on the amber device fault LEDs that make up the entity supplied. If a device name is given, the device's LED is lit. If a storageset name is given, all device LEDs that make up the storageset are lit. If a unit number is given, all device LEDs that make up the unit are lit. See LOCATE CANCEL to turn off the LEDs.
An error is displayed if no entity by that name or number has been configured.

Examples

`HSZ20>` **`LOCATE DISK0`**
Turns on the device fault LED on device DISK0.

`HSZ20>` **`LOCATE D102`**
Turns on the device fault LEDs on all devices that make up disk unit number 102.

`HSZ20>` **`LOCATE DISKS`**
Turns on the device fault LEDs on all configured disk devices.

MIRROR *disk-device-name1 container-name*

The MIRROR *disk-device-name1 container-name* command allows you to convert a physical device specified by *disk-device-name* to a mirrorset with one member. The mirrorset name is specified by *container-name*. This command can be used on devices that are already members of higher level containers (stripesets or units).

After you convert the device to a mirrorset, increase the nominal number of members with the SET *mirrorset-container-name* MEMBERSHIP=*number-of-members* command. Use the SET *mirrorset-container-name* REPLACE=*disk-device-name* command to actually add more members to the mirrorset. Refer to SET *mirror-container-name* for details for using the MEMBERSHIP=*number-of-members* qualifier versus the REPLACE=*disk-device-name* qualifier.

NOTE

When this command is used to create mirrorsets from stripeset members that were created before version 2.6 firmware, the stripeset will then be incompatible with prior firmware versions.

Under certain circumstances, use of the MIRROR command can result in a change in the geometry (that is, the number of cylinders, tracks per cylinder, and sectors per track) reported by the affected host unit. If the MIRROR command is used to convert an active, simple stripeset into a striped mirrorset, the host unit based on that stripeset can report a different geometry after the conversion than it reported prior to it. Unit capacity and data are retained; but the geometry may change.

Most file systems and applications are insensitive to changes in LUN geometry. On such operating systems, MIRROR can be used on all types of storagesets. If the data client is sensitive to unit geometry, however, or if this sensitivity is unknown, avoid the use of MIRROR on simple stripesets.

Syntax

MIRROR *DISK-DEVICE-NAME1 CONTAINER-NAME*

Parameters

disk-device-name1
Specifies the name of the physical device that you wish to convert to a one member mirrorset. The device must be part of a unit.

container-name
Specifies the name that is used to refer to this mirrorset. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up pf letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (__), for a total of nine characters.

Qualifiers

COPY=copy_speed
The COPY qualifier allows you to specify the speed at which mirrorset copies are performed. You may specify either NORMAL or FAST.
NORMAL uses relatively few controller resources to perform the copy, and has little impact on controller performance.
FAST uses more controller resources, which reduces the time it takes to complete the copy, but also reduces overall controller performance.

POLICY=BEST_FIT\POLICY=BEST_PERFORMANCE\NOPOLICY (Default)
The POLICY qualifier specifies the replacement policy to be used when a mirrorset member within the mirrorset fails.
BEST_FIT gives highest priority to finding a replacement device within the spareset that most closely matches the sizes of the remaining members of the mirrorset. If more than one device in the spareset is the correct size, the device that gives the best performance is selected.
BEST_PERFORMANCE (default) gives highest priority to finding a replacement device within the spareset that results in the best performance of the mirrorset (the device should be on a different port). If more than one device in the spareset has the best performance, the device that most closely matches the size of the remaining members of the mirrorset is selected.
NOPOLICY retries a failing device from the mirrorset without selecting a replacement. This causes the mirrorset to run with less than the nominal number of members until a BEST_FIT or BEST_PERFORMANCE policy is selected, or a member is manually replaced in the mirrorset.

Examples

```
HSZ20> MIRROR DISK210 MIRROR5
```

Creates a one member mirrorset from a DISK210 (a single disk).

REDUCE *disk-device-name1 [disk-device-nameN]*

Allows you to remove members from an existing mirrorset. This command allows you to remove members from mirrorsets. For a consistent copy of a stripeset whose members are mirrorsets, all mirrorsets must be reduced at the same time with one command. It is similar to the SET *mirrorset-container-name* REMOVE=*disk-device-name* command, except that the nominal number of members in the mirrorset is decreased by the number of members removed, and the devices are not placed in the failedset.

NOTE

Mirrorsets have both a nominal number of members and an actual number of members. The nominal number of members in a mirrorset is the number given in the SET *mirrorset-name* MEMBERSHIP=*number-of-members* command. If devices have not been added to the mirrorset, or if a member is removed, the actual number of members may be less than the nominal number. The actual number of members can never be greater than the nominal number of members.

The disk devices to be removed need not be members of the same mirrorset. The devices MUST be part of the same unit (for example, the same stripeset). This is an atomic operation because the I/O to the unit associated with the given mirrorset members is stalled while the specified mirrorset members are removed. No auto-sparing occurs and each mirrorset membership is set to the new reduced number of members. The removed devices specified by *disk-device-name1* through *disk-device-nameN* are not placed in the failedset, but are left as unused devices.

Note that for each mirrorset that you reduce, the mirrorset must have at least one remaining NORMAL member after the reduction. If this is not true for all *disk-devices-names* specified, then none of the specified mirrorsets are reduced.

NOTE

A NORMAL member is a mirrorset member whose entire contents are guaranteed to be the same as all other NORMAL members. All NORMAL members have exactly the same contents.

Syntax

REDUCE *disk-device-name1 [disk-device-name N]*

Parameters

disk-device-name1
Specifies the name of the NORMAL mirrorset member to be removed.

[disk-device-nameN]
Specifies the name of the *nth* mirrorset member to be removed.

Examples

```
HSZ20> REDUCE DISK210 DISK110
```
DISK210 and DISK110 are removed from their respective mirrorsets.

RENAME

Renames a container.

Syntax

RENAME *OLD-CONTAINER-NAME NEW-CONTAINER-NAME*

Parameters

old-container-name-
Specifies the existing name that identifies the container.

new-container-name
Specifies the new name to identify the container. This name is referred to when creating units and storagesets. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up of letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

Examples

HSZ20> **RENAME DISK0 DISK100**
Renames container DISK0 to DISK100.

RESTART THIS_CONTROLLER

Restarts this controller.

The RESTART THIS_CONTROLLER command flushes all user data from this controller's write-back cache (if present), then restarts this controller.

If any user data cannot be flushed to disk, the controller does not restart unless the IGNORE_ERRORS qualifier is specified.

Specifying IMMEDIATE causes this controller to restart immediately without flushing any user data to the disks, even if drives are online to a host.

The controller restarts and resumes operations where it was interrupted.

> **NOTE**
>
> If you enter the RESTART THIS_CONTROLLER command and you are using a virtual terminal to communicate with the controller, the connection is lost when the controller restarts.

Syntax

RESTART THIS_CONTROLLER

Qualifiers

IGNORE_ERRORS\NOIGNORE_ERRORS (Default)
If errors result when trying to write user data, the controller is not restarted unless IGNORE_ERROR is specified.

If the IGNORE_ERRORS qualifier is specified, the controller restarts even if all customer data cannot be written to disk from the write-back cache.

> **NOTE**
>
> Use of the IGNORE_ERRORS qualifier may cause a delay in saving customer data from the cache to disk. If a hardware occurs while unwritten data is in the cache, the data may be lost or corrupted.

IMMEDIATE_SHUTDOWN\NOIMMEDIATE_SHUTDOWN (Default)
If IMMEDIATE is specified, the controller is immediately restarted without checking for online devices or flushing user data from write-back cache to disk.

> **CAUTION**
>
> Use of the IMMEDIATE_SHUTDOWN qualifier causes a delay in saving customer data from the cache to disk. If a hardware fault occurs while unwritten data is in the cache, the data may be lost or corrupted.

Examples

```
HSZ20> RESTART THIS_CONTROLLER
```
Restarts this controller as long as this controller does not have any units that are online.

RETRY_ERRORS UNWRITEABLE_DATA

Tries to write the unwriteable data on a unit.

If a container fails in a way that customer data in the write-back cache cannot be written to the container, the unwriteable data error is reported. If possible the condition that is causing the unwriteable data should be corrected and the write operation should be attempted again. RETRY_ERRORS UNWRITEABLE_DATA attempts to write the unwriteable data error. No data is lost if the retry fails.

Syntax

`RETRY_ERRORS UNWRITEABLE_DATA` ***`UNIT-NUMBER`***

Parameters

unit-number
Specifies the logical unit number (D0–D7, D100-D107, and so forth) which the write operation of the unwriteable data is attempted. The *unit-number* is the name given the unit when it was created using the ADD UNIT command.

Examples

`HSZ20>` **`RETRY_ERRORS UNWRITEABLE_DATA D103`**
Attempts to write the cached data on disk unit D103 that was previously marked unwriteable.

RUN

The RUN command starts various diagnostics and utilities on THIS_CONTROLLER. Diagnostics and utilities can be run *only* on the controller where the terminal or virtual terminal connection is connected.

Syntax

RUN ***PROGRAM-NAME***

Parameters

program-name
The name of the diagnostic or utility to be run. DILX and CLONE are examples of utilities and diagnostics that can be run from the CLI. The following is a complete list of the utilities that you execute with the run command and the chapter of this document that discusses that utility (for general descriptions of each of these utilities see Chapter 2 of this Guide):

CONFIG	Chapter 3
CFMENU	Chapter 3
CLONE	Chapter 4
CLCP	Chapter 4
FLS	Chapter 4
FMU	Chapter 5
VTDPY	Chapter 5
DILX	Chapter 5
HSUTIL	Chapter 6

Examples

```
HSZ20> RUN DILX

DISK INLINE EXERCISER - VERSION 2.0
                .
                .
                .
RUNS THE DILX DIAGNOSTIC.
```

SELFTEST THIS_CONTROLLER

Runs a self-test on this controller.

The SELFTEST THIS_CONTROLLER command flushes all user data from this controller's write-back cache (if present), shuts down this controller, then restarts it in DAEMON loop-on-self-test mode. The OCP reset (//) button must be pressed to take this controller out of loop-on-self-test mode.

If any user data cannot be flushed to disk, the controller does not self-test unless the IGNORE_ERRORS qualifier is specified.

Specifying IMMEDIATE causes this controller to self-test immediately without flushing any user data to the disks, even if drives are online to a host.

NOTE

If you enter a SELFTEST THIS_CONTROLLER command, and you are using a virtual terminal to communicate with the controller, the connection is lost when this controller starts the self-test.

Syntax

SELFTEST THIS_CONTROLLER

Qualifiers

IGNORE_ERRORS\NOIGNORE_ERRORS (Default)

If errors result when trying to write user data, the controller does not start the self-test unless IGNORE_ERRORS is specified.

If the IGNORE_ERRORS qualifier is specified, the controller starts the self-test even if all customer data cannot be written to disk from the write-back cache.

NOTE

use of the IGNORE_ERRORS qualifier may cause a delay in saving user data from the cache to disk. If a hardware fault occurs while the cache contains unwritten data, the data may be lost or corrupted.

IMMEDIATE\NOIMMEDIATE (Default)

If IMMEDIATE is specified, the controller will immediately start self-test without checking for online devices or flushing user data from write cache to disk.

NOTE

Use of the IMMEDIATE qualifier causes a delay in saving customer data from the cache to disk. If a hardware fault occurs while the cache contains unwritten data, the data may be lost or corrupted.

Examples

`HSZ20>` **`SELFTEST THIS_CONTROLLER`**
Starts the self-test on this controller as long as this controller does not have any units online.

SET *disk-container-name*

Changes the transportable characteristics of a disk drive.

Syntax

```
SET DISK-CONTAINER-NAME
```

Parameters

disk-container-name
The name of the disk drive that will have its characteristics changed.

Qualifiers

TRANSPORTABLE\NOTRANSPORTABLE (Default)
In normal operations, the controller makes a small portion of the disk inaccessible to the host and uses this area to store metadata, which improves data reliability, error detection, and recovery. This vast improvement comes at the expense of transportability.
If NOTRANSPORTABLE is specified and there is no valid metadata on the unit, the unit must be initialized.

NOTE

Digital recommends that you avoid specifying TRANSPORTABLE unless transportability of disk drive or media is imperative and there is no other way to accomplish moving the data.

Examples

```
HSZ20> SET DISK130 TRANSPORTABLE
```

Sets DISK130 to transportable.

SET *FAILEDSET*

Sets the AutoNewSpare mode of the failedset.

The SET FAILEDSET command enables or disables the AutoNewSpare feature of the subsystem failedset. You can view the current state of the AutoNewSpare feature using the SHOW FAILEDSET command.

Syntax

SET *FAILEDSET AUTOSPARE/NOAUTOSPARE*

Parameters

AUTOSPARE/NOAUTOSPARE
Use AUTOSPARE to enable the AutoNewSpare feature. With AutoNewSpare enabled, the controller automatically initializes as a spare device any disk device that you install in the subsystem cabinet to replace a device in the failedset. The controller then adds it to the spareset. If the initialization fails, the controller moves the device to the failedset.
Do not use a device that has already been initialized (has metadata on it) to replace a device in the faildset. You must install the replacement device in the same slot from which you removed the failed device.

Use NOAUTOSPARE to disable the AutoNewSpare feature. With AutoNewSpare disabled, the controller leaves in the failedset any disk device that you install in the subsystem cabinet to replace a device in the failedset.

Examples

HSZ20> **SET FAILEDSET AUTOSPARE**
Enables the AutoNewSpare feature.

HSZ20> **SET FAILEDSET NOAUTOSPARE**
Disables the AutoNewSpare feature.

SET *mirrorset-container-name*

Changes the characteristics of a mirrorset.

Syntax

SET *MIRRORSET-CONTAINER-NAME*

Parameters

mirrorset-container-name
The name of the mirrorset that will have its characteristics modified. The name must start with a letter (A–Z) and can then consist of up to eight more characters made up of letters A–Z, numbers 0–9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

Qualifiers

MEMBERSHIP=number-of-members
Allows you to increase or decrease the nominal membership of a mirrorset to the number specified by *number-of-members*.

If the mirrorset membership is increased, and auto-sparing is turned on (by specifying POLICY=BEST_FIT or POLICY=BEST_PERFORMANCE), the mirrorset automatically brings in spares until either the new number of members is reached, or there are no more suitable spares.

If auto-sparing is turned off (by specifying NOPOLICY), then you must use the SET *mirrorset-container-name* REPLACE=*disk-device-name* command to bring the mirrorset up to the new nominal number of members.

You may not set the nominal number of members to be lower than the number of members physically present. Use the REMOVE=*disk-container-name* qualifier to reduce the number of devices that are part of the mirrorset.

REPLACE=disk-device-name
Specifies the replacement of a disk member into an existing mirrorset provided the following two conditions are met:

- The replacement policy must be set to NOPOLICY.
- The mirrorset must be missing at least one member.

If these two conditions are met, the device specified by *disk-device-name* is added to the mirrorset specified by *mirrorset-container-name*. The nominal number of members does not change.

NOTE

No other qualifiers to the SET *mirrorset-device-name* command may be specified if the REPLACE qualifier is specified.

REMOVE=disk-container-name

The SET *mirrorset-container-name* REMOVE=*disk-device-name* CLI command allows you to remove members from an existing mirrorset. The device specified by *disk-device-name* is removed from the mirrorset specified by *mirrorset-container-name.* If the physical device is not a member of the mirrorset, or if the mirrorset will not have a remaining NORMAL or NORMALIZING member, then an error is reported and no action is taken. On successful removal, the removed device is added to the failedset and a new member is auto-spared into the mirrorset (if applicable).

> **NOTE**
>
> A NORMAL member is a mirrorset member whose entire contents is guaranteed to be the same as all other NORMAL members. All NORMAL members have exactly the same contents.
>
> NORMALIZING members only exist when a mirrorset is first created. One member is identified as NORMAL, and all other *original* mirrorset members are marked as NORMALIZING. All new data that is written to the mirrorset is written to all members. All data on the NORMAL member that existed before the mirrorset was created is copied to the NORMALIZING members. When all the blocks on the members are the same, the NORMALIZING members are marked as NORMAL.
>
> (Members may also be marked as NORMALIZING if cache data is lost.)

Note that the nominal number of members in the mirrorset does not change. If auto-sparing does not occur, then when an acceptable spare becomes available or when the replacement policy changes, the mirrorset automatically adds the spare.

> **NOTE**
>
> No other qualifiers to the SET *mirrorset-device-name* command may be specified if the REMOVE qualifier is specified.

POLICY=BEST_FIT\POLICY=BEST_PERFORMANCE (Default)\NOPOLICY

The SET *mirrorset-container-name* POLICY=*policy-type* command specifies the replacement policy to be used when a mirrorset member within the mirrorset fails.

BEST_FIT gives highest priority to finding a replacement device within the spareset that most closely matches the sizes of the remaining members of the mirrorset. If more than one device in the spareset is the correct size, the device that gives the best performance is selected.

BEST_PERFORMANCE (default) gives highest priority to finding a replacement device within the spareset that results in the best performance of the mirrorset (the device should be on a different port). If more than one device in the spareset has the best performance, the device that most closely matches the size of the remaining members of the mirrorset is selected.

NOPOLICY retries a failing device from the mirrorset without selecting a replacement. This causes the mirrorset to run with less than the nominal number of members until a BEST_FIT or BEST_PERFORMANCE policy is selected, or a member is manually replaced in the mirrorset.

READ_SOURCE=read-source

The SET *mirrorset-container-name* READ_SOURCE=*read-source* command allows you to control the read algorithm for the specified mirrorset. The following choices are allowed for read-source:

ROUND_ROBIN—Each NORMAL mirrorset member is the target of a read in sequential membership order. No preference is given to any NORMAL member. This is the default READ_SOURCE.

LEAST_BUSY—The NORMAL mirrorset member with the least busy work queue is the target of the read.

device-container-name—All reads are done on *device-container-name.* If *device-container-name* fails out of the mirrorset, the READ_SOURCE algorithm reverts to LEAST_BUSY.

COPY=copy_speed

The SET *mirrorset-container-name* COPY=*copy_speed* command allows you to specify the speed at which mirrorset copies are performed. You may specify either NORMAL or FAST.

NORMAL uses relatively few controller resources to perform the copy, and has little impact on controller performance.

FAST uses more controller resources, which reduces the time it takes to complete the copy, but also reduces overall controller performance.

Examples

`HSZ20>` **`SET MIRR9 POLICY=BEST_FIT`**
Changes mirrorset MIRR9's policy to BEST_FIT.

`HSZ20>` **`SET MIRR9 REMOVE=DISK0`**
Removes mirrorset MIRR9's member DISK0 from the mirrorset. If there is a replacement policy, a new disk is taken from the spareset and placed in the mirrorset automatically.

`HSZ20>` **`SET MIRR9 REPLACE=DISK320`**
Adds disk DISK320 to the reduced mirrorset, MIRR9. A reconstruct operation begins immediately on DISK320.

SET *RAIDset-container-name*

Changes the characteristics of a RAIDset.

Syntax

SET ***RAIDSET-CONTAINER-NAME***

Parameters

RAIDset-container-name
The name of the RAIDset that will have its characteristics modified.

Qualifiers

POLICY=BEST_FIT\POLICY=BEST_PERFORMANCE (Default)\NOPOLICY

Specifies the replacement policy to use when a member within the RAIDset fails.

BEST_FIT gives highest priority to finding a replacement device within the spareset that most closely matches the sizes of the remaining members of the RAIDset. If more than one device in the spareset is the correct size, the device that gives the best performance is selected.

BEST_PERFORMANCE (default) gives highest priority to finding a replacement device within the spareset that results in the best performance of the RAIDset. If more than one device in the spareset has the best performance, the device that most closely matches the size of the remaining members of the RAIDset is selected.

NOPOLICY retires a failing device from the RAIDset without selecting a replacement. This causes the RAIDset to run in a reduced state until a BEST_FIT or BEST_PERFORMANCE policy is selected, or a member is manually replaced in the RAIDset (see SET *raidset-container-name*).

RECONSTRUCT=NORMAL (Default)

Specifies the speed at which a RAIDset will be reconstructed when a new member is added to the RAIDset or immediately after the RAIDset is initialized.

RECONSTRUCT=NORMAL (default) balances overall performance of the controller against the demand of reconstructing the RAIDset.

RECONSTRUCT=FAST (default) compromises overall performance of the controller to speed reconstructing the RAIDset.

REMOVE=disk-container-name

Specifies the removal of a disk member from a RAIDset. If the RAIDset is already in a reduced state, an error is displayed and the command is rejected. If a replacement policy is specified, the replacement is taken from the spareset to replace the removed member using the specified policy. If NOPOLICY is specified, the RAIDset continues to operate in a reduced state until a replacement is manually specified (see SET *RAIDset-container-name* REPLACE=) or a policy is specified (see SET *RAIDset-container-name* POLICY=).

The disk removed via the REMOVE= command is added to the failedset.

NOTE

No other qualifiers to the SET *RAIDset-container-name* command may be specified if REMOVE is specified.

REPLACE=disk-container-name

Specifies the replacement of a disk member into a reduced RAIDset. If the RAIDset is not in a reduced state, an error is displayed and the command is rejected. If a replacement policy is already specified, an error is displayed and the command is rejected. If the disk specified is already being used by a configuration (including a spareset), an error is displayed and the command is rejected. Otherwise, the disk specified is added as a member to the specified RAIDset and a reconstruct operation begins immediately.

NOTE

No other qualifiers to the SET *RAIDset-container-name* command may be specified if REPLACE is specified.

Examples

```
HSZ20> SET RAID9 POLICY=BEST_FIT
```

Changes RAIDset RAID9's policy to BEST_FIT.

```
HSZ20> SET RAID9 REMOVE=DISK0
```

Removes RAIDset RAID9's member DISK0 from the RAIDset. If there is a replacement policy, a new disk is taken from the spareset and placed in the RAIDset automatically.

```
HSZ20> SET RAID9 REPLACE=SPAREDISK
```

Adds disk SPAREDISK to the reduced RAIDset, RAID9. A reconstruct operation begins immediately on SPAREDISK.

SET THIS_CONTROLLER

The SET THIS_CONTROLLER command allows you to modify controller parameters on THIS_CONTROLLER, the controller connected to the maintenance terminal or virtual terminal.

Syntax

```
SET THIS_CONTROLLER
```

Qualifiers

CACHE_FLUSH_TIMER=n\CACHE_FLUSH_TIMER=DEFAULT

Specifies how many seconds (1–65535) of idle time may elapse before the write-back cache flushes its entire contents to disk. After the specified time, the write-back cache flushes its contents to disk to ensure data integrity. The default value is 10 seconds.

A controller restart is required to place changes in effect.

CACHE_POLICY=A (Default)\CACHE_POLICY=B

Allows selection of the write-back cache battery policy used by the controller. The policy affects the availability of RAIDsets and mirrorsets when the battery condition is low during controller initialization.

The parameter change takes effect immediately.

The CACHE_POLICY setting affects RAIDsets and mirrorsets as follows:

If the CACHE_POLICY on the controller is set to A and the batteries are low when the controller initializes, any RAIDset or mirrorset that does not have access to good batteries is made inoperative.

If the CACHE_POLICY on the controller is set to B and the batteries are low when the controller initializes, any RAIDset or mirrorset that does not have access to good batteries is accessed in write-through (read cache) mode.

Regardless of the CACHE_POLICY setting, a low or bad battery affects controller operation:

Stripesets and disk-based units with write-back caching enabled are accessed in write-through (read cache) mode, as long as unwritten cache data has not been lost.

If the batteries go low after controller initialization, unwritten cache data is flushed from the cache and any RAIDset or mirrorset that does not have access to good batteries is made inoperative, regardless of the cache policy.

If the batteries are bad or missing, RAIDsets and mirrorsets are made inoperative.

Write-back caching automatically resumes when the cache batteries are fully recharged or replaced.

If the cache policy is set to A, you can add mirrorsets and RAIDsets to your configuration, but you cannot access them until the batteries are fully charged.

CAUTION

There is some risk in setting CACHE_POLICY=B to allow access to mirrorsets and RAIDsets when the batteries are low. Because the batteries may be in an unknown state, there is no guarantee as to how long they will maintain data in the cache if a power failure occurs. Fully charged batteries will maintain the data for a minimum of 100 hours.

As long as the cache battery is charges and the system is operating normally, the firmware checks the condition of the battery once every 24 hours. If the battery fails or becomes discharged, the firmware checks its condition every 4 minutes.

HOST_FUNCTION=(mode)

Configures the host port for compatibility with various operating systems. Table A−1 describes available host compatibility modes. Use mode A (the default) unless you specifically need to use one of the other modes.

Table A−2 Host_Function Mode Description

Mode	Description
A (default)	Most Hosts
B	IBM AIX
C	Siemens-Nixdorf Hosts
D	Windows NT

The current setting of the Host Function Mode is displayed with the SHOW THIS_CONTROLLER and SHOW OTHER_CONTROLLER commands.

ID=(n1[,nN])

Specifies from one to four SCSI target IDs (0–7). If two or more target IDs are specified, they must be enclosed in parenthesis and separated by a comma.

NOTE

The unit number determines which target the LUN will be available under. For example, D203 would be target 2, LUN 3. D500 would be target 5, LUN 0. D5 would be target 0, LUN 5.

INITIAL_CONFIGURATION

Recovers subsystem configuration information previously saved on disk with the INITIALIZE SAVE_CONFIGURATION command. Specifying the INITIAL_CONFIGURATION parameter causes the controller to load its nonvolatile memory with the configuration information saved on disk. The controller halts after it executes this command. You must restart the controller by pressing its reset button.

The controller displays an error message if it cannot find any saved configuration information.

You must specify this parameter completely and cannot abbreviate it.

PROMPT="new prompt"

Specifies a 1- to 16-character prompt enclosed in quotes that will be displayed when the controller's CLI prompts for input. Only printable ASCII characters are valid.

When first installed, the CLI prompt is set to the controller's model number (for example, HSZ20>).

TERMINAL_PARITY=ODD\TERMINAL_PARITY=EVEN\NOTERMINAL_PARITY

Specifies the parity transmitted and expected. Parity options are ODD or EVEN.

NOTERMINAL_PARITY causes the controller to not check for, or transmit any parity on the terminal lines.

When first installed, the controller's terminal parity is set to NOTERMINAL_PARITY.

TERMINAL_SPEED=baud_rate

Sets the terminal speed to 300, 600, 1200, 2400, 4800, 9600 or 19200 baud. The transmit speed is always equal to the receive speed.

When first installed, the controller's terminal speed is set to 9600 baud.

TIME=dd-mmm-yyyy:hh:mm:ss

The TIME= command specifies the date and time.

Examples

```
HSZ20> SET THIS_CONTROLLER TERMINAL_SPEED=1200
```
Sets the terminal speed to 1200 baud.

```
HSZ20> SET THIS_CONTROLLER ID=5
```
Sets THIS controller so it responds to requests for target 5.

```
HSZ20> SET THIS_CONTROLLER ID=(2,5)
```
Sets THIS controller so it responds to requests for targets 2 and 5.

```
HSZ20> SET THIS_CONTROLLER HOST_FUNCTION=?
Your options are:
   A
   B
   C
   D
```
Lists the available Host Function Mode options.

```
HSZ20> SET THIS_CONTROLLER HOST_FUNCTION=A
```
Sets this controller to host function mode A.

SET *unit-number*

Changes the unit parameters.

Syntax

SET *UNIT-NUMBER*

Parameters

unit-number
Specifies the logical unit number (D0–D7, D100-D107, and so forth) to modify the software switches. The *unit-number* is the name given the unit when it was created using the ADD UNIT command.

Qualifiers for a Unit Created from a TRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

READ_CACHE (Default)\NOREAD_CACHE
Enables and disables the controller's read cache on this unit.

RUN (Default)\NORUN

Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

Qualifiers for a Unit Created from a NOTRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

READ_CACHE (Default)\NOREAD_CACHE
Enables and disables the controller's read cache on this unit.

RUN (Default)\NORUN
Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

WRITEBACK_CACHE\NOWRITEBACK_CACHE (Default)
Enables and disables the controller's write-back cache on this unit.

NOTE

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Qualifiers for a Unit Created from a RAIDset

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

RUN (Default)\NORUN

Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)

Enables and disables write protection of the unit.

NOTE

Writes may still be performed to a write-protected RAIDset to satisfy a reconstruct pass or to reconstruct a newly replaced member. However, write protect will disable the writing of any new data.

WRITEBACK_CACHE\NOWRITEBACK_CACHE (Default)
Enables and disables the controller's write-back cache on this unit.

NOTE

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Qualifiers for a Unit Created from a Stripeset

MAXIMUM_CACHED_TRANSFER=n\ MAXIMUM_CACHED_TRANSFER=32 (Default)
Specifies the maximum size transfer in blocks to be cached by the controller. Any transfers over this size are not cached. Valid values are 1–1024.

READ_CACHE (Default)\NOREAD_CACHE
Enables and disables the controller's read cache on this unit.

RUN (Default)\NORUN

Enables and disables a unit's availability to the host. When RUN (default) is specified, the devices that make up the unit will be spun up and the unit will be made available to the host. If NORUN is specified, the devices that make up the unit will still be spun up, but the unit will not be made available to the host.

WRITE_PROTECT\NOWRITE_PROTECT (Default)
Enables and disables write protection of the unit.

WRITEBACK_CACHE\NOWRITEBACK_CACHE (Default)
Enables and disables the controller's write-back cache on this unit.

NOTE

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Examples

```
HSZ20> SET D1 WRITE_PROTECT NOREAD_CACHE
```

Sets the write protect and turns off the read cache on unit D1.

SHOW DEVICES

Shows all physical devices known to the controller and physical device information.

Syntax

SHOW DEVICES

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each device. Information contained in the additional information is dependent on the device type.

Examples

```
HSZ20> SHOW DEVICES
NAME          TYPE          PORT TARG  LUN              USED BY
-----------------------------------------------------------------
DI0           DISK             1    0    0              D100
DI1           DISK             1    1    0              D110
```

Shows a basic listing of devices attached to the controller.

```
HSZ20> SHOW DEVICES FULL \
NAME          TYPE          PORT TARG  LUN               USED BY
-----------------------------------------------------------------
DI0           DISK             1    0    0               D100
           DEC      RZ35     (C) DEC X388
DI1           DISK             1    1    0               D110
           DEC      RZ26     (C) DEC T386
```

Shows a full listing of devices attached to the controller.

SHOW DISKS

Shows all disk drives and drive information known to the controller.

Syntax

SHOW DISKS

Qualifiers

FULL
If the FULL qualifier is specified, additional information may be displayed after each device.

Examples

```
HSZ20> SHOW DISKS
NAME          TYPE         PORT TARG  LUN              USED BY
 --------------------------------------------------------------
DI0           DISK            1    0    0              D100
DI1           DISK            1    1    0              D110
```

Shows a basic listing of disks attached to the controller.

```
HSZ20> SHOW DISKS FULL
NAME          TYPE         PORT TARG  LUN               USED BY
 --------------------------------------------------------------
DI0           DISK            1    0    0               D100
           DEC     RZ26     (C) DEC X388
         SWITCHES:
           NOTRANSPORTABLE
         SIZE: 2050353 BLOCKS
         CONFIGURATION BEING BACKED UP ON THIS CONTAINER
DI1           DISK            1    1    0               D110
           DEC     RZ26     (C) DEC T386
         SWITCHES:
           NOTRANSPORTABLE
         SIZE: 2050353 BLOCKS
```

Shows a full listing of disks attached to the controller.

SHOW *disk-container-name*

Shows information about a specific particular disk.

Syntax

SHOW *DISK-CONTAINER-NAME*

Parameters

disk-container-name
The name of the disk drive to be displayed.

Examples

```
HSZ20> SHOW D130
NAME          TYPE          PORT TARG  LUN               USED BY
 ----------------------------------------------------------------
DI3           DISK             1    3    0               D130
           DEC      RZ26     (C) DEC X388
         SWITCHES:
           NOTRANSPORTABLE
         SIZE: 2050353 BLOCKS
```

Shows a listing of disk D130.

SHOW FAILEDSET

The SHOW FAILEDSET command displays all the disk drives that are members of the failedset. It also displays the current status of the AutoNewSpare feature.

A disk drive that the firmware moves into a failedset is automatically spun down. Its fault light flashes, as long as it remains in the failedset. The controller signals that a device on the associated port has failed by illuminating that port's fault LED

Syntax

SHOW FAILEDSET

Examples

```
HSZ20> SHOW FAILEDSET
NAME          STORAGESET                    USES        USED BY
------------------------------------------------------------------
FAILEDSET     FAILEDSET                     DISK110
                                            DISK200
        SWITCHES:
          AUTOSPARE
```

Shows a listing of the members of the failedset.

SHOW MIRRORSETS

Shows all configured mirrorsets known to the controller and any mirrorset-specific data related only to mirrorsets.

Syntax

SHOW MIRRORSETS

Qualifiers

FULL
If the FULL qualifier is specified, additional information may be displayed after each mirrorset.

Examples

```
HSZ20> SHOW MIRRORSETS
NAME       STORAGESET          USES                   USED BY
---------------------------------------------------------------
MIRR1      MIRRORSET           DISK100                STR0
                               DISK200
MIRR2      MIRRORSET           DISK110                STR0
                               DISK210
MIRR3      MIRRORSET           DISK120                STR0
                               DISK220
```

MIRR1, MIRR2, and MIRR3 are all two-member mirrorsets. All three mirrorsets are members of stripeset STR0.

```
HSZ20>SHOW MIRRORSETS FULL
NAME          STORAGESET                     USES        USED BY
 ----------------------------------------------------------------
MIRR1         MIRRORSET                      DISK100     STR0
                                             DISK200
                   SWITCHES:
                     POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
                     COPY (PRIORITY) = NORMAL
                     READ_SOURCE = LEAST_BUSY
                     MEMBERSHIP = 2, 2 MEMBERS PRESENT
                   STATE:
                     DISK100   (MEMBER  0) IS NORMAL
                     DISK200   (MEMBER  1) IS NORMAL
                   SIZE: 2050353 BLOCKS
MIRR2         MIRRORSET                      DISK110     STR0
                                             DISK210
                   SWITCHES:
                     POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
                     COPY (PRIORITY) = FAST
                     READ_SOURCE = LEAST_BUSY
                     MEMBERSHIP = 2, 2 MEMBERS PRESENT
                   STATE:
                     DISK110   (MEMBER  0) IS NORMAL
                     DISK210   (MEMBER  1) IS NORMAL
                   SIZE: 4109470 BLOCKS
MIRR3         MIRRORSET                      DISK120     STR0
                                             DISK220
                   SWITCHES:
                     POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
                     COPY (PRIORITY) = FAST
                     READ_SOURCE = LEAST_BUSY
                     MEMBERSHIP = 2, 2 MEMBERS PRESENT
                   STATE:
                     DISK110   (MEMBER  0) IS NORMAL
                     DISK210   (MEMBER  1) IS NORMAL
                   SIZE: 4109470 BLOCKS
```

Shows extended information for all mirrorsets known to the controller.

SHOW *mirrorset-container-name*

Shows the same information as SHOW MIRRORSETS FULL except that it only displays information on the one mirrorset specified.

Syntax

SHOW *MIRRORSET-CONTAINER-NAME*

mirrorset-container-name
The name of the mirrorset to be displayed.

Examples

```
HSZ20>SHOW MIRR0
NAME          STORAGESET                    USES        USED BY
 ----------------------------------------------------------------
MIRR0         MIRRORSET                     DISK130     D200
         SWITCHES:
           NOPOLICY (FOR REPLACEMENT)
           COPY (PRIORITY) = NORMAL
           READ_SOURCE = LEAST_BUSY
           MEMBERSHIP = 2, 1 MEMBER PRESENT
         STATE:
           DISK130   (MEMBER  0) IS NORMAL
         SIZE: 4109470 BLOCKS
```

Shows a complete listing of the mirrorset named MIRR0.

SHOW RAIDSETS

Displays all the RAIDsets known by the controller and RAIDset information.

Syntax

SHOW RAIDSETS

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each storageset.

Examples

```
HSZ20> SHOW RAIDSETS
NAME          STORAGESET                    USES         USED BY
----------------------------------------------------------------
R0            RAIDSET                       DISK100      D100
                                            DISK110
                                            DISK200
R1            RAIDSET                       DISK120      D101
                                            DISK210
                                            DISK220
```

Shows a basic listing of all RAIDsets.

```
HSZ20> SHOW RAIDSETS FULL
NAME          STORAGESET                    USES         USED BY
----------------------------------------------------------------
R0            RAIDSET                       DISK100      D100
                                            DISK110
                                            DISK200
         SWITCHES:
           POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
           RECONSTRUCT (PRIORITY) = NORMAL
           CHUNKSIZE = 63 BLOCKS
         STATE:
           RECONSTRUCT 3% COMPLETE
           DISK100   (MEMBER  0) IS RECONSTRUCTING
           DISK110   (MEMBER  1) IS RECONSTRUCTING
           DISK200   (MEMBER  2) IS RECONSTRUCTING
         SIZE: 2050353 BLOCKS
R1            RAIDSET                       DISK120      D101
                                            DISK210
                                            DISK220
         SWITCHES:
           POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
           RECONSTRUCT (PRIORITY) = NORMAL
           CHUNKSIZE = 63 BLOCKS
         STATE:
           NORMAL
           DISK120   (MEMBER  0) IS NORMAL
           DISK210   (MEMBER  1) IS NORMAL
           DISK220   (MEMBER  2) IS NORMAL
         SIZE: 2050353 BLOCKS
```

Shows a full listing of all RAIDsets.

SHOW *raidset-container-name*

Shows specific information about a particular RAIDset.

Syntax

SHOW *RAIDSET-CONTAINER-NAME*

Parameters

raidset-container-name
The name of the RAIDset to be displayed.

Examples

```
HSZ20> SHOW RAID9
NAME          STORAGESET                        USES        USED BY
-----------------------------------------------------------------
RAID9         RAIDSET                           DISK100
                                                DISK110
                                                DISK200
                                                DISK210
         SWITCHES:
           POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
           RECONSTRUCT (PRIORITY) = NORMAL
           CHUNKSIZE = 63 BLOCKS
         STATE:
           NORMAL
           DISK100   (MEMBER  0) IS NORMAL
           DISK110   (MEMBER  1) IS NORMAL
           DISK200   (MEMBER  2) IS NORMAL
           DISK210   (MEMBER  3) IS NORMAL
         SIZE: 2050353 BLOCKS
```

Shows a listing of RAIDset RAID9.

SHOW SPARESET

Displays all the disk drives that are members of the spareset.

Syntax

SHOW SPARESET

Examples

```
HSZ20> SHOW SPARESET
NAME          STORAGESET                     USES        USED BY
-----------------------------------------------------------------
SPARESET      SPARESET                       DISK130
```

Shows a list of the members of the spareset.

SHOW STORAGESETS

The SHOW STORAGESETS command displays all the storagesets known by the controller and information about them. A storageset is any collection of containers, such as stripesets, mirrorsets, RAIDsets, the spareset and the failedset.

Syntax

SHOW STORAGESETS

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each storageset.

Examples

```
HSZ20> SHOW STORAGESETS
NAME          STORAGESET                    USES        USED BY
-----------------------------------------------------------------
S0            STRIPESET                     DISK100        D100
                                            DISK110
                                            DISK220
```

Shows a basic listing of all storagesets.

```
HSZ20> SHOW STORAGESETS FULL
NAME          STORAGESET                    USES       USED BY
----------------------------------------------------------------
M3            MIRRORSET                     DISK120       D100
                                            DISK130
        SWITCHES:
          POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
          COPY (PRIORITY) = NORMAL
          READ_SOURCE = LEAST_BUSY
          MEMBERSHIP = 2, 2 MEMBERS PRESENT
        STATE:
          DISK120   (MEMBER  0) IS NORMAL
          DISK130   (MEMBER  1) IS NORMAL
        SIZE: 4109470 BLOCKS
R1            RAIDSET                       DISK100       D105
                                            DISK110
                                            DISK200
        SWITCHES:
          POLICY (FOR REPLACEMENT) = BEST_PERFORMANCE
          RECONSTRUCT (PRIORITY) = NORMAL
          CHUNKSIZE = 256 BLOCKS
        STATE:
          RECONSTRUCT 19% COMPLETE
          DISK100   (MEMBER  0) IS NORMAL
          DISK110   (MEMBER  1) IS NORMAL
          DISK200   (MEMBER  2) IS NORMAL
        SIZE: 4099696 BLOCKS
SPARESET      SPARESET                      DISK210
FAILEDSET     FAILEDSET
        SWITCHES:
          NOAUTOSPARE
```

Shows a full listing of all storagesets.

SHOW STRIPESETS

Displays all the stripesets known by the controller and related stripeset information.

Syntax

SHOW STRIPESETS

Qualifiers

FULL
If the FULL qualifier is specified, additional information may be displayed after each storageset.

Examples

```
HSZ20> SHOW STRIPESETS
NAME          STORAGESET                     USES        USED BY
----------------------------------------------------------------
S0            STRIPESET                      DISK100     D100
                                             DISK110
                                             DISK200
S1            STRIPESET                      DISK120     D200
                                             DISK210
                                             DISK220
```

Shows a basic listing of all stripesets.

```
HSZ20> SHOW STRIPESETS FULL
NAME          STORAGESET                     USES        USED BY
-----------------------------------------------------------------
S0            STRIPESET                      DISK100     D100
                                             DISK110
                                             DISK200
         SWITCHES:
           CHUNKSIZE = 24 BLOCKS
         STATE:
            NORMAL
           DISK100   (MEMBER  0) IS NORMAL
           DISK110   (MEMBER  1) IS NORMAL
           DISK200   (MEMBER  2) IS NORMAL
         SIZE: 2050353 BLOCKS
S1            STRIPESET                      DISK120     D200
                                             DISK210
                                             DISK220
         SWITCHES:
           CHUNKSIZE = 24 BLOCKS
         STATE:
           NORMAL
           DISK120   (MEMBER  0) IS NORMAL
           DISK210   (MEMBER  1) IS NORMAL
           DISK220   (MEMBER  1) IS NORMAL
         SIZE: 2050353 BLOCKS
```

Shows a full listing of all stripesets.

SHOW *stripeset-container-name*

Displays specific information about a particular stripeset.

Syntax

SHOW *STRIPESET-CONTAINER-NAME*

stripeset-container-name
The name of the stripeset to be displayed.

Examples

```
HSZ20> SHOW STRIPE0
NAME          STORAGESET                    USES        USED BY
-----------------------------------------------------------------
S0            STRIPESET                     DISK100     D100
                                            DISK110
                                            DISK200
         SWITCHES:
           CHUNKSIZE = 24 BLOCKS
         STATE:
           NORMAL
           DISK100   (MEMBER  0) IS NORMAL
           DISK110   (MEMBER  1) IS NORMAL
           DISK200   (MEMBER  2) IS NORMAL
         SIZE: 2050353 BLOCKS
```

Shows a listing of stripeset STRIPE0.

SHOW THIS_CONTROLLER

Shows all controller, port, and terminal information for this controller.

Syntax

SHOW THIS_CONTROLLER

Qualifiers

FULL
If the FULL qualifier is specified, additional information is displayed after the basic controller information.

Examples

```
HSZ20> SHOW THIS_CONTROLLER
CONTROLLER:
CONTROLLER:
        SC4200 CX54100219 FIRMWARE V27Z-0, HARDWARE  A02
        NOT CONFIGURED FOR DUAL-REDUNDANCY
        SCSI ADDRESS 7
        TIME: NOT SET
HOST PORT:
        SCSI TARGET(S) (1, 2), NO PREFERRED TARGETS
CACHE:
      16 MEGABYTE WRITE CACHE, VERSION 2
      CACHE IS GOOD
      BATTERY IS GOOD
      NO UNFLUSHED DATA IN CACHE
      CACHE_FLUSH_TIMER = DEFAULT (10 SECONDS)
      CACHE_POLICY = A
      HOST FUNCTIONALITY MODE = A
LICENSING INFORMATION:
      RAID (RAID OPTION) IS ENABLED, LICENSE KEY IS VALID
      WBCA (WRITEBACK CACHE OPTION) IS ENABLED, LICENSE KEY IS
VALID
      MIRR (DISK MIRRORING OPTION) IS ENABLED, LICENSE KEY IS
VALID
EXTENDED INFORMATION
      TERMINAL SPEED 19200 BAUD, EIGHT BIT, NO PARITY, 1 STOP
BIT
      OPERATION CONTROL:  00000004  SECURITY STATE CODE:  83462
      CONFIGURATION BACKUP ENABLED ON 2 DEVICES
```

Shows the complete controller Information.

SHOW UNITS

Shows all units known by the controller and unit information.

Syntax

SHOW UNITS

Qualifiers

FULL

If the FULL qualifier is specified after UNITS, additional information may be displayed after each unit-number, such as the switch settings.

Examples

```
HSZ20> SHOW UNITS
     LUN                                     USES
-------------------------------------------------------------
   D100                                      M3
   D105                                      R1
```

Shows a basic listing of units available on the controller.

```
HSZ20> SHOW UNITS FULL
     LUN                                     USES
----------------------------------------------------------------
   D100                                      M3
         SWITCHES:
           RUN                NOWRITE_PROTECT        READ_CACHE
           NOWRITEBACK_CACHE
           MAXIMUM_CACHED_TRANSFER_SIZE = 32
         STATE:
           ONLINE to this controller
           NO EXCLUSIVE ACCESS
           PREFERRED_PATH = THIS_CONTROLLER
         SIZE: 4109470 BLOCKS
   D105                                      R1
         SWITCHES:
           RUN                NOWRITE_PROTECT        READ_CACHE
           NOWRITEBACK_CACHE
           MAXIMUM_CACHED_TRANSFER_SIZE = 32
         STATE:
           ONLINE to this controller
           NO EXCLUSIVE ACCESS
           PREFERRED_PATH = THIS_CONTROLLER
         SIZE: 4099696 BLOCKS
```

Shows a full listing of units available on the controller.

SHOW UNIT NUMBER

Displays specific information about a particular unit

Syntax

SHOW *UNIT-NUMBER*

Parameters

unit-number
The unit number of the unit that is to be displayed.

Parameters

```
HSZ20> SHOW D100
     LUN                                        USES --------------
------------------------------------
   D100                                         M3
         SWITCHES:
           RUN                NOWRITE_PROTECT        READ_CACHE
           NOWRITEBACK_CACHE
           MAXIMUM_CACHED_TRANSFER_SIZE = 32
         STATE:
           ONLINE to this controller
           NO EXCLUSIVE ACCESS
           PREFERRED_PATH = THIS_CONTROLLER
         SIZE: 4109470 BLOCKS
```

Shows a listing of a specific disk unit.

SHUTDOWN THIS_CONTROLLER

Shuts down and does not restart this controller.

Syntax

SHUTDOWN THIS_CONTROLLER

Description

The SHUTDOWN THIS_CONTROLLER command flushes all user data from this controller's write-back cache (if present), then shuts down this controller.
If any user data cannot be flushed to disk, the controller does not shut down unless the IGNORE_ERRORS qualifier is specified.

Specifying IMMEDIATE causes this controller to shut down immediately without flushing any user data to the disks, even if drives are online to a host.

The shutdown may take a long time if the controller has active unit I/O.

> **NOTE**
>
> If you issue a SHUTDOWN THIS_CONTROLLER command, communication with the controller is lost when this controller shuts down.

Qualifiers

IGNORE_ERRORS\NOIGNORE_ERRORS (Default)

If errors result when trying to write user data, the controller is not shut down unless IGNORE_ERROR is specified.

If the IGNORE_ERRORS qualifier is specified, the controller shuts down even if all customer data cannot be written to disk from the write-back cache.

> **NOTE**
>
> The IGNORE_ERRORS qualifier may cause a delay in saving user data from the cache to disk. If a hardware fault occurs while unwritten data is in the cache, the data may be lost or corrupted.

IMMEDIATE_SHUTDOWN\NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE is specified, the controller shuts down immediately without checking for online devices or flushing user data from write-back cache to disk.

> **NOTE**
>
> The IMMEDIAGE_SHUTDOWN qualifier may cause a delay in saving user data from the cache to disk. If a hardware fault occurs while unwritten data is in the cache, the data may be lost or corrupted.

Examples

```
HSZ20> SHUTDOWN THIS_CONTROLLER
```

Shuts down this controller as long as this controller.

UNMIRROR *disk-device-name*

Converts a one member mirrorset back to a single device and deletes the mirrorset from the list of known mirrorsets.

Allows you to convert a mirrorset with one disk, specified by the disk name, *disk-device-name*, to a physical device. This command can be used on mirrorsets that are already members of higher level containers (stripesets or units).

You can use the UNMIRROR command only on a device with a capacity that matches that of the mirrorset of which it is a member. If a device has a larger capacity than the mirrorset of which it is a member, the controller displays an error message indicating the unmirror command is invalid.

Syntax

UNMIRROR ***DISK-DEVICE-NAME***

Examples

```
HSZ20> UNMIRROR DISK130
```

Converts a DISK130 back to a single device. An error message prints if the disk is not a mirrorset.

A.2 CLI Messages

The following sections describes the messages you can encounter during interactive use of the CLI.

A.2.1 Error Conventions

An Error *nnnn:* message means that the command did not complete. Multiple error messages may result from one command.

A.2.2 CLI Error Messages

Following are the error messages that you may encounter while using the CLI:

Message:
`Error 1000: The LUN portion of the unit number must be from 0 to 7`
Explanation: This error results from an ADD UNIT command when the *n* in the D*n* or T*n* specified is out of range.

Retry the ADD UNIT command with a correct number.

Message:
`Error 1010: Maximum cached transfer size must be 1 through 1024 blocks`
Explanation: This error results from a SET <unit number> or an ADD UNIT command when MAXIMUM_CACHED_TRANSFER_SIZE was specified.
MAXIMUM_CACHED_TRANSFER_SIZE must be in the range 1 through 1024.

Retry the SET or ADD command with a correct number.

Message:
`Error 1020: CHUNKSIZE must be from <minimum> to <maximum>`
Explanation: This error results from an INITIALIZE *storageset-container-name* command when CHUNKSIZE was specified. The chunk size must be DEFAULT or greater than 15. Retry the INITIALIZE command with DEFAULT or a correct number.

Message:
`Error 1100: Disk unit numbers must start with the letter 'D'`
Explanation: All disk unit numbers are of the form “Dn”. This error is displayed if you add a disk unit that does not begin the unit number with the letter “D.”

Retry the ADD command with the letter “D” at the beginning of the unit number.

Message:
`Error 1110: Unit numbers may not have leading zeros`
Explanation: Disk unit numbers may not be of the form "D03," for example, "D3" should be specified.

Retry the ADD command without any leading zeros.

Message:
`Error 1120: LUN <lun> is already used`
Explanation: Lun number <lun> has already been used by a disk.

Retry the ADD command specifying a different LUN.

Message:
`Error 1130: The unit number cannot exceed <max unit>`
Explanation: You specified a unit number that was out-of-bounds.

Try to add the unit again using a unit number that is less than or equal to <max unit>.

Message:
`Error 1140: Invalid unit number. Valid unit number range(s) are: <start> to <end>`
Explanation: You attempted to create a unit out of the valid unit ranges. The valid unit ranges are given by the <start> and <end> values.

Retry the ADD command specifying a unit number in the correct range.

Message:
`Error 1150: A restart of THIS_CONTROLLER is required before units may be added`
Explanation: You changed the target IDs that THIS_CONTROLLER supports without restarting the controller, then tried to add a unit that is supported by the new target IDs. Before the new target ids may be used, a restart is required.

Restart the controller.

Message:
`Error 1160: A restart of OTHER_CONTROLLER is required before units may be added`
Explanation: You changed the target IDs that the OTHER_CONTROLLER supports without restarting the controller, then tried to add a unit that is supported by the new target IDs. Before the new target ids may be used, a restart is required.

Restart the controller.

Message:
`Error 2000: Port must be 1 - <maximum port number>`
Explanation: When adding a device, you specified a port less than 1 or greater than <maximum port number>.

Retry the command specifying a port within the range given.

Message:
`Error 2010: Target must be 0 - <maximum target number>`
Explanation: When adding a device, you specified a target greater than <maximum target number>.

<maximum target number> is 6.

Message:
`Error 2020: LUN must be 0 - 7`
Explanation: When adding a device, you specified a LUN greater than 7.

Message:
`Error 2030: This port, target LUN combination already in use by another device`
Explanation: When adding a device, you specified PTL that is already specified by another device.

Message:
`Error 2040: Cannot set TRANSPORTABLE when device in use by an upper layer`
Explanation: A disk cannot be set to TRANSPORTABLE when it is being used by an upper level (unit or storageset).

Message:
`Error 2050: Cannot set NOTRANSPORTABLE when device in use by an upper layer`
Explanation: A disk cannot be set to NOTRANSPORTABLE when it is being used by an upper level (unit or storageset).

Message:
`Error 2060: Can only clear UNKNOWN errors on a device`
Explanation: You attempted to clear UNKNOWN on a storageset or a unit.

Check the name of the device and reissue the command.

Message:
`Error 3020: Cannot add <disk name>; This will cause too many members in MIRRORSET <mirrorset name>. Use a SET <mirrorset name> MEMBERSHIP=<new count> to command increase the membership count first`
Explanation: You can not add more disks to a mirrorset than is specified by the mirrorset's membership count. In order to successfully do a SET <mirrorset name> REPLACE=<disk name>, to a "full" mirrorset, you must first increase the mirrorset count.

Message:
`Error 3030: <disk name> is not a member of <mirrorset name>, cannot be used for the read source`
Explanation: When specifying a specific disk as a read source for a mirrorset, the disk specified must be a member of the mirrorset.

Choose a disk that's a member of the mirrorset and retry the command.

Message:
`Error 3040: Cannot set read source to a specific device if mirrorset not configured as a unit`
Explanation: Setting a mirrorset's read source to a specific device cannot be accomplished if the mirrorset is not configured as a unit.

Create a unit from the mirrorset and try the command again.

Message:
`Error 3050: <disk name> could not be initialized as a spare disk`
Explanation: When adding spare disks to the spareset, they are initialized with special spare disk metadata. If the metadata cannot be written, error 3050 results.

Message:
`Error 3060: <disk name> is not a member of the spareset`
Explanation: You attempted to delete a disk drive from the spareset that was not a member of the spareset.

Message:
`Error 3070: <disk name> is not a member of the failedset`
Explanation: You attempted to delete a disk drive from the failedset that was not a member of the failedset.

Message:
`Error 3080: <setname> can't be deleted`
Explanation: You attempted to delete the spareset or the failedset. These containers cannot be deleted.

Message:
`Error 3090: <licensable feature> support is not enabled on this controller`
Explanation: You attempted to use a feature that requires a license, and the license was not enabled on this controller.

Message:
`Error 3100: <licensable feature> support is not enabled on other controller`
Explanation: You attempted to use a feature that requires a license, and the license was not enabled on the other controller.

Message:
`Error 3110: <disk name> is not a member of <container name>, cannot remove it`
Explanation: When issuing a SET <container name> REMOVE=<disk name>, the disk specified was not part of the container.

Check the device and container names and reissue the command.

Message:
`Error 3120: <container name> is already reduced. Another member cannot be removed`
Explanation: When issuing a SET <container name> REMOVE=<disk name>, the container was already in a reduced state. Add another disk before removing another member.

Message:
`Error 3130: Unable to remove <disk name> from <container name>`
Explanation: When issuing a SET <container name> REMOVE=<disk name>, the controller was unable to remove the device from the RAIDset.

Check for error conditions, and if none exist, contact Digital Multivendor Customer Services.

Message:
`Error 3140: <disk name> is in a spareset. Remove it from the spareset first.`
Explanation: When issuing a SET <container name> REPLACE=<disk name>, the disk specified was part of the spareset. A disk to be used as a replacement must not be part of any configuration.

Message:
`Error 3150: <disk name> is still part of a configuration. Delete upper configuration first.`
Explanation: When issuing a SET <container name> REPLACE=<disk name>, the disk specified was part of an existing configuration. A disk to be used as a replacement must not be part of any configuration.

Message:
`Error 3170: <container name> is not reduced. Cannot replace a member`
Explanation: When issuing a SET <container name> REPLACE=<disk name>, the container specified was not reduced.

Remove a member before replacing it.

Message:
`Error 3180: <container name> has a replacement policy specified. Cannot manually replace a member.`
Explanation: When issuing a SET <container name> REPLACE=<disk name>, it was discovered that the container specified already had a replacement policy specified. A manual replacement cannot be done on a container with an automatic replacement policy.

Set the replacement policy for the container to NOPOLICY and try the replacement again.

Message:
`Error 3190: Unable to replace <disk name> in <container name>`
Explanation: When issuing a SET <container name> REPLACE=<disk name>, the controller was unable to replace the device into the RAIDset.

Check for error conditions, and if none exist, contact Digital Multivendor Customer Services.

Message:
`Error 3200: No other switches may be specified on a REMOVE operation.`
Explanation: When issuing a SET <container name> REMOVE=<disk name>, no other switches (such as POLICY) may be specified.

Message:
`Error 3210: No other switches may be specified on a REPLACE operation.`
Explanation: When issuing a SET <container name> REPLACE=<disk name>, no other switches (such as POLICY) may be specified.

Message:
`Error 3220: A REPLACE may not be done on a raidset or mirrorset that is not configured as a unit`
Explanation: A REPLACE operation may not be done on a RAIDset or Mirrorset that has not been configured as a unit.

Message:
`Error 3230: <container name> is reconstructing <disk name>. Only <disk name> may be removed`
Explanation: When issuing a SET <container name> REMOVE=<disk name> on a RAIDset that is already reconstructing, only the disk drive that is being reconstructed may be removed.

Message:
`Error 3240: <storageset type> may not be initialized`
Explanation: Sparesets and failedsets cannot be initialized.

Check the name of the container that you wish to initialize and try again.

Message:
`Error 3250: A REMOVE may not be done on a raidset or mirrorset that is not configured as a unit`
Explanation: A RAIDset or mirrorset must be configured as a unit before a disk may be removed.

Create a unit from the RAIDset or Mirrorset and then remove the member.

Message:
`Error 3260: <disk name> is a TRANSPORTABLE disk. TRANSPORTABLE disks cannot be used by storagesets. Do a SET <disk name> NOTRANSPORTABLE before using this disk in a storageset`
Explanation: You cannot place a TRANSPORTABLE disk into a reduced RAIDset.

Set the disk NOTRANSPORTABLE and retry the command.

Message:
`Error 3270: <disk name> not in NORMAL state. Only NORMAL state units may be specified as a read source`
Explanation: You may not specify a Mirrorset member as a read source unless it's in NORMAL state.

Either wait for the desired disk to enter NORMAL state or choose another disk in the Mirrorset that's already in NORMAL state.

Message:
`Error 3280: Cannot determine if <disk name> is in the NORMAL state`
Explanation: Unable to determine at this point in time if the disk specified is in normal state.

Retry the command. If this error persists, call Digital Multivendor Customer Services.

Message:
`Error 3290: Mirrorsets are limited to 6 members maximum`
Explanation: A Mirrorset may only have 6 disks as members. If you try to exceed this amount the above message is printed.

Message:
`Error 3300: Currently there are <current count> members in this mirrorset. You must specify a membership greater than or equal to <current count>.`
Explanation: You attempted to set the number of Mirrorset members to less than the actual number of disk drives that make up this Mirrorset.

Either specify a number greater than or equal to the number of physical disk drives that make up this mirrorset or remove one or more disk drives, then reduce the member count.

Message:
`Error 3310: No other switches may be specified on a MEMBERSHIP operation`
Explanation: When specifying the number of members in a mirrorset, no other switches may be specified.

Retry the command only specifying the membership count.

Message:
`Error 3320: Unable to change membership on mirrorset <mirrorset name>`
Explanation: Unable to set the membership count on the specified mirrorset at this point in time.

Retry the command. If this error persists, call Digital Multivendor Customer Services.

Message:
`Error 3330: Can only MIRROR disks. <disk name> is not a disk`
Explanation: Only disks may be mirrored. If you specify a device other than a disk drive, the above error is printed.

Try the command again specifying a disk drive.

Message:
`Error 3340: Can only UNMIRROR disks. <disk name> is not a disk`
Explanation: Only disks may be unmirrored. If you specify a device other than a disk drive, the above error is printed.

Try the command again specifying a disk drive.

Message:
`Error 3350 <disk name> must be configured under a mirrorset to UNMIRROR`
Explanation: Only disk drives that are configured under mirrorsets may be unmirrored.

Specify a disk that is configured under a mirrorset.

Message:
`Error 3360: To UNMIRROR a disk, the mirrorset must contain only one member, the disk to be UNMIRROR. Mirrorset <mirrorset name> contains more than one member`
Explanation: To unmirror a disk drive, only one disk may be a member of the mirrorset.

REMOVE all disk drives but one and retry the command.

Message:
`Error 3370: A MIRROR may not be done on a disk that is not configured as a unit`
Explanation: To MIRROR a disk drive either it or the storageset that it belongs to must be configured as a unit.

Configure the disk drive or the storageset that it belongs to as a unit and retry the command.

Message:
`Error 3380: An UNMIRROR may not be done on a disk that is not configured as a unit`
Explanation: To UNMIRROR a disk drive either it or the storageset that it belongs to must be configured as a unit.

Configure the disk drive or the storageset that it belongs to as a unit and retry the command.

Message:
`Error 3390: REDUCE of specified disks failed`
Explanation: The requested REDUCE did not complete successfully.

Check the configuration to assure that it was not partially completed (some disks removed from the specified mirrorset(s); if so, reconfigure and wait for all disks to return to the NORMAL state. If this error persists, call Digital Multivendor Customer Services.

Message:
`Error 3400: <device name> is not a disk. Can only REDUCE disks`
Explanation: Only disk drives may be specified on the REDUCE command.

Retry the command only specifying disks.

Message:
`Error 3410: <disk name> was specified twice in the REDUCE command`
Explanation: The disk drive name called out was specified twice in the REDUCE command line.

Retry the command with only specifying disk names once.

Message:
`Error 3420: <disk name> is not part of the same unit as previous disks specified`
Explanation: All disks specified on the REDUCE command must be used by one common unit. If the disks specified are in use by more than one unit, the above message is printed.

Retry the command specifying disks that all have one common unit as a parent.

Message:
`Error 3430: A REDUCE may not be done on disks not configured as a unit`
Explanation: A REDUCE command is only valid when the storageset is configured as a unit.

Configure the top most storageset as a unit and retry the command.

Message:
`Error 3440: <disk name> is not a member of a mirrorset`
Explanation: The disk drive specified is not a member of a mirrorset.

Retry the command only specifying disks that are members of Mirrorsets.

Message:
`Error 3450: No NORMAL or NORMALIZING members would be left in MIRRORSET <mirrorset name>`
Explanation: When reducing, at least one member that is in the NORMAL state must remain in each Mirrorset. This error results when the Mirrorset called out would not be left with at least one NORMAL member.

Make sure at least on NORMAL member will remain in each Mirrorset specified and retry the command.

Message:
`Error 3460: !AC cannot be mirrored due to bad cache on <controller> controller`
Explanation: In order to mirror a disk drive the writeback cache must be correctly operating on the controller. If it's not, the above error results.

Repair the cache on the problem controller(s) and retry the command.

Message:
`Error 3470: Only 20 mirrorsets and raidsets total can be supported on a controller. This storageset not added.`
Explanation: The total number of RAIDsets and Mirrorsets supported on the controller is 20; this error will result if you attempt to add an 21st MIRROR or RAIDset.

Delete unused RAID and Mirrorsets and retry the command.

Message:
`Error 3480: Only 30 stripesets, mirrorsets and raidsets total can be supported on a controller. This storageset not added.`
Explanation: The total number of RAIDsets, Mirrorsets and Stripesets supported on the controller is 30; this error will result if you attempt to add an 21st MIRROR, STRIPE or RAIDset.

Delete unused RAID, STRIPE and Mirrorsets and retry the command.

Message:
`Error 4000: The CLI prompt must have 1 to 16 characters.`
Explanation: This error results from a SET THIS_CONTROLLER or SET OTHER_CONTROLLER command with the qualifier PROMPT=. The length of the CLI prompt must be at least one character and may not exceed 16 characters.

Retry the command with the correct number of characters.

Message:
`Error 4010: Illegal character in CLI prompt.`
Explanation: A nonprintable character was specified. Only ASCII characters space “ ” through tilde “~” may be specified (hex 20–7E).

Message:
`Error 4020: Terminal speed must be 300, 1200, 2400, 4800, 9600 or 19200`
Explanation: This error results from a SET THIS_CONTROLLER or SET OTHER_CONTROLLER command with the argument TERMINAL_SPEED=. The only valid baud rates that may be specified are 300, 1200, 2400, 4800, 9600 or 19200 baud.

Retry the command with a correct terminal speed.

Message:
`Error 4090: Module has invalid serial number. This controller cannot be used. Call field service`
Explanation: This error is typically the result of faulty Non-Volatile memory. This error cannot be fixed in the field.

A replacement controller must be ordered. Contact Digital Multivendor Customer Services.

Message:
`Error 4100: Unable to RESTART other controller.`
Explanation: A communication error occurred when trying to restart the other controller.

Retry the RESTART command.

Message:
`Error 4110: Unable to SHUTDOWN other controller.`
Explanation: A communication error occurred when trying to shutdown the other controller.

Retry the SHUTDOWN command.

Message:
`Error 4120: Unable to SELFTEST other controller.`
Explanation: A communication error occurred when trying to self-test the other controller.

Retry the SELFTEST command.

Message:
`Error 4130: Unable to setup controller restart.`
Explanation: A communication error occurred when trying to restart or self-test the other controller.

Retry the RESTART or SELFTEST command.

Message:
`Error 4150: Unable to rundown the following units on the other controller: <list of problem units>`
Explanation: When attempting to shut-down, restart or selftest the other controller, some units could not be successfully run down. This can be caused by errors when trying to rundown the units. Rectify the problems on the problem units and reissue the SHUTDOWN, RESTART or SELFTEST command.

Message:
`Error 4160: Unable to rundown the following units on this controller: <list of problem units>`
Explanation: When attempting to SHUTDOWN, RESTART or SELFTEST this controller, some units could not be successfully run down. This can be caused either by online units or errors when trying to rundown the units.

Rectify the problems on the problem units and reissue the SHUTDOWN, RESTART or SELFTEST command.

Message:
`Error 4170: Only <max targets> targets may be specified`
Explanation: When setting THIS_CONTROLLER or OTHER_CONTROLLER ID=, you specified too many IDs; you can only specify up to <max targets> IDs.

Retry the SET THIS_CONTROLLER ID= command with no more than <max targets> IDs specified.

Message:
`Error 4180: Invalid unit number(s) still present that must be deleted before the controller ID may be changed. All unit numbers must be in the range(s): <start> to <end>`
Explanation: You attempted to change the controller IDs when there were still units using those IDs. The current valid unit ranges are given by the <start> and <end> values.

Either delete the units that use the ID that will no longer be specified, or Retry the SET THIS_CONTROLLER ID= specifying the ID being used by the existing units.

Message:
`Error 4190: The time must be specified in the format dd-mmm-yyyy:hh:mm:ss`
Explanation: The time must be specified as shown.

Retry the command using the correct time format.

Message:
`Error 4200: CACHE_FLUSH_TIMER must be in the range 1 to 65535`
Explanation: The value given for the CACHE_FLUSH_TIMER is out of range.

Reissue the command specifying a number in the range shown.

Message:
`Error 4210: Only targets defined by the ID= command may be specified on the PREFERRED_ID= qualifier`
Explanation: The PREFERRED_IDs specified must be a subset of the IDs (targets) supported by the controller. When changing either the supported targets or the preferred ids, it was found that the PREFERRED_IDs were not a subset of the IDs.

Reissue the command with valid PREFERRED_ID= arguments or change the IDs supported by the controller.

Message:
`Error 5000: A program name must be from 1 to 6 characters in length`
Explanation: This error results from a “RUN <program name>.”

Message:
`Error 5010: The requested program is currently busy.`
Explanation: This error results from a “RUN <program name>.” The program requested is being run by someone else.

Message:
`Error 5020: The requested program is unknown.`
Explanation: This error results from a "RUN <program name>."

Enter "DIR" to get a list of available programs.

Message:
`Error 5030: Insufficient memory for request.`
Explanation: This error results from a "RUN <program name>" resource problem. Retry the command later.

Message:
`Error 6000: Communication failure with the other controller.`
Explanation: There was a communication problem with the other controller. This typically happens if the other controller is shutting down. If these messages happen often when the other controller is not shutting down, call Digital Multivendor Customer Services.

Message:
`Error 7000: Can only clear LOST_DATA cache errors on a unit.`
Explanation: You specified something other than a unit for clearing the LOST_DATA cache error.

Message:
`Error 7010: Can only clear UNWRITEABLE_DATA cache errors on a unit.`
Explanation: You specified something other than a unit for clearing the UNWRITEABLE_DATA cache error.

Message:
`Error 7020: Can only retry UNWRITEABLE_DATA cache errors on a unit`
Explanation: You specified something other than a unit for retrying a write on a UNWRITEABLE_DATA cache error.

Message:
`Error 7030: Unable to force write of unwriteable data`
Explanation: A RETRY UNWRITEABLE_DATA command could not write the UNWRITEABLE_DATA.

Message:
`Error 7040: Unable to rundown unit before clearing error`
Explanation: To clear UNWRITEABLE_DATA and LOST_DATA errors, the unit must be rundown before the error is cleared. If the unit could not be rundown, the above error results. If this error persists, call Digital Multivendor Customer Services.

Message:
`Error 7050: Unable to runup unit after clearing error. This controller must be restarted`
Explanation: To clear UNWRITEABLE_DATA and LOST_DATA errors, the unit must be rundown before the error is cleared. If the unit was rundown and the error was cleared and then the unit was unable to be run back up, the unit will remain unavailable until the controller is restarted.

Message:
`Error 7060: Cannot clear LOST_DATA on a unit without LOST_DATA`
Explanation: If a unit does not have a LOST_DATA error, a CLEAR LOST_DATA <unit number> is an illegal command.

Check to assure the unit that you wished to CLEAR LOST_DATA on does exhibit a LOST_DATA error.

Message:
`Error 7070: Cannot clear UNWRITEABLE_DATA on a unit without UNWRITEABLE_DATA`
Explanation: If a unit does not have a UNWRITEABLE_DATA error, a CLEAR UNWRITEABLE_DATA <unit number> is an illegal command.

Check to assure the unit that you wished to CLEAR UNWRITEABLE_DATA on does exhibit an UNWRITEABLE_DATA error.

Message:
`Error 7080: Cannot retry UNWRITEABLE_DATA on a unit without UNWRITEABLE_DATA`
Explanation: If a unit does not have a UNWRITEABLE_DATA error, a RETRY UNWRITEABLE_DATA <unit number> is an illegal command.

Check to assure the unit that you wished to RETRY UNWRITEABLE_DATA on does exhibit a UNWRITEABLE_DATA error.

Message:
`Error 9000: Cannot rename a unit`
Explanation: Only devices and storagesets may be renamed. If you attempt to rename a unit, the above message results.

Message:
`Error 9010: <name> is an illegal name, it must be from 1 to 9 characters.`
Explanation: This error results from an ADD command with an illegal name given.

Message:
`Error 9020: <name> is an illegal name, it must start with A-Z`
Explanation: This error results from an ADD command with an illegal name given.

Message:
`Error 9030: <name> is an illegal name, characters may consist only of A-Z, 0-9, ., - or _`
Explanation: This error results from an ADD command with an illegal name given.

Message:
`Error 9040: <name> conflicts with keyword <keyword>`
Explanation: The name given in an ADD command conflicts with a CLI keyword.

Specify another name.

Message:
`Error 9050: Configuration area full`
Explanation: The total number of units, devices, and storagesets that can be configured is 195 in any combination. This error results when you exceed that number of nodes.

Delete some units or devices in order to recover some configuration nodes.

Message:
`Error 9060: <name> does not exist`
Explanation: Some operation (SET, DELETE, INITIALIZE, and so forth) specified a name that does not exist.

Check the name and retry the command.

Message:
`Error 9070: <name> is part of a configuration`
Explanation: Devices may not be deleted if they are still in use by storagesets or units. Storagesets may not be deleted if they are still used by units.

Delete configurations from the top down; delete units, then stripesets, and RAIDsets (if any), and then finally devices.

Message:
`Error 9080: <name> is already used`
Explanation: An ADD command specified a name that is already in use.

Specify another name.

> **NOTE**
>
> This error is commonly the result of failing to name a storageset. For example, when trying to create a stripeset out of three disks the user may type:
>
> ADD STRIPESET DISK100 DISK200 DISK300

This will result in a 9080 error since the controller attempted to create a stripeset named "DISK100" (since the user forgot to name the stripeset), which of course is a name already used by a disk drive.

Message:
`Error 9100: A <storageset type> must have from <minimum> to <maximum> entities`
Explanation: The wrong number of devices was specified for this storageset. Different storagesets require different numbers of devices.

Reexamine the configuration, then correct the number of devices.

Message:
`Error 9150: INITIALIZE is no longer supported at the unit level. You must INITIALIZE the container that makes up this unit`
Explanation: You tried to initialize a unit. Units cannot be initialized. The container that makes up the unit must be initialized before a unit is created out of the container.

Message:
`Error 9170: <device type> <device name> at PTL <port> <target> <lun> No device installed`
Explanation: When a unit is added or initialized, the configuration of the devices that makes up the unit is checked. If no device is found at the PTL specified, this error is displayed.

Check both the logical and physical configuration of the unit and correct any mismatches.

Message:
`Error 9200: <name> conflicts with unit names`
Explanation: This error results from an ADD command. Names in the format of D*n* and T*n*, when *n* is a number from 0 to 4094, are reserved for units. Rename the storageset or device that is being added so it does not conflict with the unit names and retry the command.

Message:
`Error 9210: Cannot check if drives are online to the other controller`
Explanation: When trying to check for online drives on the other controller, there was a communication failure.

Retry the command.

Message:
`Error 9220: You cannot specify NOREAD_CACHE while in WRITEBACK_CACHE mode`
Explanation: WRITEBACK_CACHE is specified for this unit. READ_CACHE cannot be specified while WRITEBACK_CACAHE is set.

Reissue the SET command disabling both WRITEBACK_CACHE and READ_CACHE.

Message:
`Error 9230: Unable to modify switches requested`
Explanation: This error results from a SET command. The system is currently busy.

Retry the SET command later.

Message:
`Error 9240: Cannot delete unit in maintenance mode`
Explanation: When trying to delete a unit, the unit was found to be in maintenance mode. This is typically the result of trying to delete a unit that is in use by DILX or TILX.

Ensure that DILX and TILX is not being run against the unit that is to be deleted, and retry the command.

Message:
`Error 9250: Initialize of disk failed`
Explanation: Unable to write metadata on disk.

Make sure the disk is not broken.

Message:
`Error 9260: Cannot INITIALIZE a container that is still part of a configuration. Delete upper configuration first`
Explanation: A container cannot be initialized that is part of another configuration or is being used by a unit.

Delete the upper configuration and reissue the INITIALIZE command.

Message:
`Error 9280: Cannot rename the SPARESET or FAILEDSET`

Message:
`Error 9310: No metadata found on container, unit not created. <reason for failure>`
Explanation: You attempted to create a unit from a container that did not have valid metadata.

INITIALIZE the metadata on the container, then create a unit out if it.

Message:
`Error 9330: NV memory write collision.  Please try again`
Explanation: Two processes were trying to modify the controller's configuration at the same time.

Check the configuration you were trying to modify to make sure it's unchanged and retry the command.

Message:
`Error 9340: Reduced raidsets cannot be INITIALIZED`
Explanation: You cannot INITIALIZE a RAIDset that is running in reduced state.

Replace a member and try again.

Message:
`Error 9380: Unable to allocate unit for NORUN to RUN transition`
Explanation: The unit could not be allocated so the controller could do a RUN/NORUN transition.

Retry the command. If this error persists, call Digital Multivendor Customer Services.

Message:
`Error 9400: Cannot rundown or allocate unit in order to delete it`
Explanation: Retry the command. If this error persists, call Digital Multivendor Customer Services.

Message:
`Error 9410: Cannot delete unit –<type> error exists on unit that must be cleared first.  To clear error type: <clear error string>`
Explanation: Units cannot be deleted if cache errors exist. Any cache errors must be cleared before a unit can be deleted.

Issue the <clear error string> command and then delete the unit.

Message:
`Error 9450: Cannot mirror <disk name>`
Explanation: Error when attempting to mirror a disk drive.

Check the configuration to assure it's the same as before. If so, retry the command. If this error persists, call Digital Multivendor Customer Services.

A.3 Warning Messages

The following sections describe the warning messages conventions and the actual warning messages that you may encounter while using the CLI and how to respond.

A.3.1 Conventions

A Warning *nnnn:* message means that the command completed, but there is a situation that you should be aware of. Typically, but not always, a warning will result in an unusable configuration; you will have to either logically reconfigure the cabinet using the CLI or physically reconfigure the cabinet by moving the disks around.

Multiple warning messages may result from one command.

Items in angle brackets (<>) are replaced at run time with names, numbers, and so on.

A.3.2 CLI Warning Messages

The following are the warning messages you may encounter while using the CLI.

Message:
```
Warning 1000: It is recommended that you read the controller product
documentation to understand the significance of enabling WRITEBACK_CACHE
particularly for RAID Arrays
```
Explanation: Using write-back cache introduces behaviors that you should completely understand before using. See the full documentation on write-back cache in the user guide.

Message:
```
Warning 3000: This storageset is configured with more than one disk per
port.  This causes a degradation in performance
```
Explanation: This warning results from an ADD *storageset-type* command. The storageset specified has more than one member per port. One method of increasing the controller's performance is through parallel transfers to members of a storageset. If multiple members of a storageset are on one port, transfers must be done in serial to those members.

Though multiple storageset members on one port will work, it is strongly recommended that the storageset be deleted and reconfigured with one member per port.

Message:
`Warning 3020: <storageset name> is configured with different sized containers.  This will result in a storageset of reduced size`
Explanation: This warning results from an ADD *storageset-type* command. Storageset size is determined by the size of the smallest device, so the storageset configured will be of reduced size.

If a reduced size storageset is acceptable, nothing needs to be done in response to the above warning. To realize the maximum storageset size, the size of all devices that make up the storageset should be identical.

Message:
`Warning 3030: Cannot determine state of disk <disk name>in mirrorset <mirrorset name>`
Explanation: Before disk drives are removed from a MIRRORset, the MIRRORset is checked to assure that at least one NORMAL member will be left after the REMOVE or REDUCE. If the state of a disk cannot be determined, the above message is printed.

Message:
`Warning 3040: <storageset name> will be disabled due to bad cache state on <controller> controller`
Explanation: If the cache is in a state on this or the other controller such that the storageset cannot be used, it will be created but the above message will be printed so the user knows that the storageset will be disabled.

Message:
`Warning 4000: A restart of this controller is required before all the parameters modified will take effect`
Explanation: This warning results from a SET THIS_CONTROLLER command. Some controller parameters require a restart before they can take effect. If any of those parameters are changed, this warning is displayed.

It is recommended that a restart via the “RESTART THIS_CONTROLLER” command be done as soon as possible.

Message:
`Warning 4010: A restart of the other controller is required before all the parameters modified will take effect`
Explanation: This warning results from a SET OTHER_CONTROLLER command. Some controller parameters require a restart before they can take effect. If any of those parameters are changed, this warning is displayed.

Restart the controller and retry the command.

Message:
`Warning 4020: A restart of both this and the other controller is required before all the parameters modified will take effect`
Explanation: This warning results from a SET THIS_CONTROLLER or a SET OTHER_CONTROLLER command. Some controller parameters require a restart of both controllers before they can take effect. If any of those parameters are changed, this warning is displayed. Restart both controllers and retry the command.

Message:
`Warning 7000: Data written successfully before clearing unwriteable data error`
Explanation: As a result of a CLEAR UNWRITEABLE_DATA, if the last-ditch attempt to write data before clearing the error was successful, the above warning is displayed.

> **NOTE**
>
> This means that no customer data was lost, so this warning is actually good.

Message:
`Warning 7010: Unable to clear LOST_DATA on other controller`
Explanation: When trying to clear LOST_DATA on the other controller, a communication error occurred.

Retry the command. If the failure persists, contact Digital Multivendor Customer Services.

Message:
`Warning 7020: Unable to clear UNWRITEABLE_DATA on other controller`
Explanation: When trying to clear UNWRITEABLE_DATA on the other controller, a communication error occurred.

Retry the command. If the failure persists, contact Digital Multivendor Customer Services.

Message:
`Warning 9000: Drive has LOST_DATA`
Explanation: During a check of the drive's metadata, it was detected that the drive had lost data.

Clear the lost data error on the drive.

Message:
`Warning 9030: Cannot determine if the correct device type is at the PTL specified`
Explanation: When a device is added, the location specified is checked to see if the correct device type is present. This warning results when no device responds from the location specified.

Check the physical configuration and the PTL that was specified.

Message:
`Warning 9040: There is currently a <device type> at the PTL specified`
Explanation: When a device is added, the location specified is checked to see if the correct device type is present. This warning results when a device different from the one specified is found at the location specified.

Check the physical configuration and the PTL that was specified.

Message:
```
Warning 9050: <device type> <device name> at PTL <port> <target> <lun>
No device installed
```
Explanation: When a unit is added, the configuration of the disks that make up the unit is checked. If no device is found at the PTL specified, this warning is displayed.

Check both the logical and physical configuration of the devices that make up the unit and correct any mismatches.

Message:
```
Warning 9060: <device type> <device name> at PTL <port> <target> <lun>
Incorrect device type installed
```
Explanation: When a unit is added, the configuration of the disks that make up the unit is checked. If a non disk device is found at the PTL specified, this warning is displayed.

Check both the logical and physical configuration of the devices that make up the unit and correct any mismatches.

Message:
```
Warning 9080: <license> support is not licensed on <controller>
controller.  Any use of this feature requires licensing.  Continued use
does not comply with the terms and conditions of licensing for this
product.
```
Explanation: You have a licensed feature enabled on this controller but it is not licensed. This is against the contractual agreement between Digital and your company. Please disable the licensed feature and contact Digital Multivendor Customer Services if you wish to purchase it.

Message:
```
Warning 9090: Metadata found on container.  Are you sure this is a
TRANSPORTABLE container?
```
Explanation: When a transportable disk was initialized, metadata was found.

Verify that this disk in fact should be marked transportable. No action is required to correct this warning.

Message:
`Warning 9100: Bad or low battery or bad write cache on <controller> writeback cache will not be used`
Explanation: The battery is low or bad on the specified controller. The unit specified will not use write-back cache until the battery is charged or repaired.

Message:
`Warning 9110: Bad or low battery or bad write cache on <controller> this unit cannot be used by <controller>`
Explanation: The battery is low or bad on the specified controller. The unit specified requires the use of write-back cache, so its use has been disabled until the battery is charged or repaired.

Index

C

D

D

E

F

H

I

L

M

O

P

R

S

T

U

V