StorageWorks™

SWXRC-04 RAID Array Controller

User's Guide

Order Number: EK–SMCS1–UG. B01

Digital Equipment Corporation
Maynard, Massachusetts

First Edition, June 1995

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

Digital Equipment Corporation does not give a warranty of any kind regarding the fitness or applicability of the information content for a particular purpose. The user assumes all responsibility for understanding the interrelationships of this enclosed information with other affected software or system products.

The disclosure of this information does not grant to the user a license under any patents, pending patents, trademarks, or copyrights or other rights of Digital Equipment Corporation, or of any third party.

FCC Notice: This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such radio frequency interference when operated in a commercial environment. Any changes or modifications made to this equipment may void the user's authority to operate the equipment. Operation of this equipment in a residential area may cause interference, in which case the user, at his own expense, may be required to take measures to correct the interference.

Copyright © Digital Equipment Corporation 1994
All Rights Reserved

Printed in U. S. A

The following are trademarks of Digital Equipment Corporation:
StorageWorks, and the Digital Logo.

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

Contents

1 Overview

2 Functional Description

5 Operating the Storage Subsystem

6 Working with RAID Arrays

7 Error Analysis and Fault Isolation

8 Diagnostics, Exercisers, and Utilities

9 Removing and Replacing Field Replaceable Units

B Command Line Interpreter

StorageWorks Glossary

Examples

Figures

Tables

About This Manual...

About This Manual describes how the content of this manual is organized.

This guide presents operation and maintenance information for the SWXRC-04 RAID controller. It presents a description of the controller and covers controller and subsystem configuration, operating, troubleshooting, and removal and replacement procedures.

The SWXRC-04 RAID controller is also called “the SWXRC-04 controller” or “the controller” in this manual.

Intended Audience

This guide is intended for user and maintenance personnel who need assistance in operating and maintaining the SWXRC-04 RAID controller.

Structure

This guide contains the following chapters:

Chapter 1	Provides an overview of the SWXRC-04 controller.
Chapter 2	Provides a technical explanation of SWXRC-04 controller hardware and firmware.
Chapter 3	Provides information to support the installation of SWXRC-04 controller storage subsystems.
Chapter 4	Defines physical configuration rules for the SWXRC-04 controller subsystem.
Chapter 5	Provides operation and logical device configuration instructions.
Chapter 6	Provides information on upgrading the SWXRC-04 controller and storage subsystem.
Chapter 7	Discusses how to translate error information and perform initial fault analysis.
Chapter 8	Details the diagnostics, inline exercisers, and utilities for the SWXRC-04 controller.
Chapter 9	Provides procedures for the removal and replacement of FRUs.
Appendix A	Lists the SWXRC-04 controller FRUs, including part numbers and related FRUs.
Appendix B	Provides complete details for CLI commands and their usage.
StorageWorks Glossary	Lists acronyms and terms specific to StorageWorks systems.

Related Documentation

Table 1 lists documents containing information related to this product.

Table 1 Related Documentation

Document Title	Order Number
Getting Started RAID Array 410 for Solaris 2.x	*EK-SMRA1-IG*
Getting Started RAID Array 410 for Solaris 1.x	*EK-SMRA2-IG*
Getting Started RAID Array 410 for HP-UX	*EK-SMRA3-IG*
Getting Started RAID Array 410 for IBM AIX 3.2.5	*EK-SMRA4-IG*
RAID Array 410 Manager for Sun (V2.5)	*AA-QEPJB-TE*
StorageWorks Solutions SWXSC-AA Office Expansion RAID Enclosure User's Guide	*EK-SMCPD-UG*
StorageWorks SW500-Series Cabinet Installation and User's Guide	*EK-SW500-IG*
StorageWorks SW800-Series Data Center Cabinet Installation and User's Guide	*EK-SW800-IG*
StorageWorks SWXSC-AA Office Expansion RAID Enclosure User's Guide	*EK-SMCPD-UG*
SWXD3-SB 2.10 GB, 3.5-inch Disk Drive SBB Product Notes	*EK-SM2DD-PN*
SWXD3-SC 1.05 GB, 3.5-inch L. P. Disk Drive SBB Product Notes	*EK-SM2DE-PN*
SWXD3-SE 4.29 GB, 3.5-inch L.P. Disk Drive SBB Product Notes	*EK-SM2DG-PN*
SWXD3-WB 2.10 GB, 3.5-inch Disk Drive SBB Product Notes	*EK-SM2DJ-PN*
SWXD3-WC 1.05 GB, 3.5-inch L.P. Disk Drive SBB Product Notes	*EK-SM2DI-PN*

Documentation Conventions

The following conventions are used in this guide:

boldface type	Boldface type in examples indicates user input. Boldface type in text indicates the first instance of terms defined in either the text, the glossary, or both.
italic type	Italic type indicates emphasis, variables in command strings, and complete manual titles.
UPPERCASE	Words in uppercase text indicate a command, the name of a file, or an abbreviation for a system privilege.
Ctrl/*x*	Ctrl/x indicates that you hold down the Ctrl key while you press another key, indicated by *x*. For DILX, the caret symbol (^) is equivalent to the Ctrl key and these same instructions apply.

Manufacturer's Declarations.

CAUTION

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

ACHTUNG !

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

ATTENTION !

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

Für Bundesrepublik Deutschland

For Federal Republic of Germany

Pour la République féderale d'Allemagne

Hochfrequenzgerätezulassung und Betriebsgenehmigung

Bescheinigung des Herstellers/Importeurs:

Hiermit wird bescheinigt, daß die Einrichtung in Übereinstimmung mit den Bestimmungen der DBP-Verfügung 523/1969, Amtsblatt 113/1969, und Grenzwertklasse "A" der VDE0871, funkenstört ist.

Das Bundesamt für Zulassungen in der Telekommunikation der Deutschen Bundespost (DBP), hat diesem Gerät eine FTZ- Serienprüfnummer zugeteilt.

Betriebsgenehmigung:

Hochfrequenzgeräte dürfen erst in Betrieb genommen werden, nachdem hierfür von dem für den vorgesehenen Aufstellungsort zuständigen Fernmeldeamt mit Funkstörungsmeßstelle die Genehmigung erteilt ist. Als Antrag auf Erteilung einer Genehmigung dient eine Anmeldepostkarte (Anhang des Handbuches) mit Angabe der FTZ-Serienprüfnummer. Der untere Teil der Postkarte ist vom Betreiber zu vervollständigen und an das örtliche Fernmeldeamt zu schicken. Der obere Teil bleibt beim Gerät.

Betreiberhinweis:

Das Gerät wurde funktechnisch sorgfältig entstört und geprüft. Die Kennzeichnung mit der Zulassungsnummer bietet lhnen die Gewähr, daß dieses Gerät keine anderen Fernmeldeanlagen einschließlich Funkanlagen stört. Sollten bei diesen Geräten ausnahmsweise trotzdem, z.B. im ungünstigsten Fall beim Zusammenschalten mit anderen EVA-Geräten, Funstörungen auftreten kann das im Einzelnen zusätzliche Funkentstörungsmaßnahmen durch den Benutzer erfordern. Bei Fragen hierzu wenden Sie sich bitte an die örtlich zuständige Funkstörungsmeßstelle lhres Fernmeldeamtes.

Externe Datenkabel:

Sollte ein Austausch der von Digital spezifizierten Datenkabel nötig werden, muß der Betreiber für eine einwandfreie Funkentstörung sicherstellen, daß Austauschkabel im Aufbau und Abschirmqualität dem Digital Originalkabel entsprechen.

Kennzeichnung:

Die Geräte werden bereits in der Fertigung mit der Zulassungsnummer gekennzeichnet und mit einer Anmeldepostkarte versehen. Sollte Kennzeichnung und Anmeldepostkarte übergangsweise nicht mit ausgeliefert werden kontaktieren Sie bitte das nächstgelegene Digital Equipment Kundendienstbüro.

1

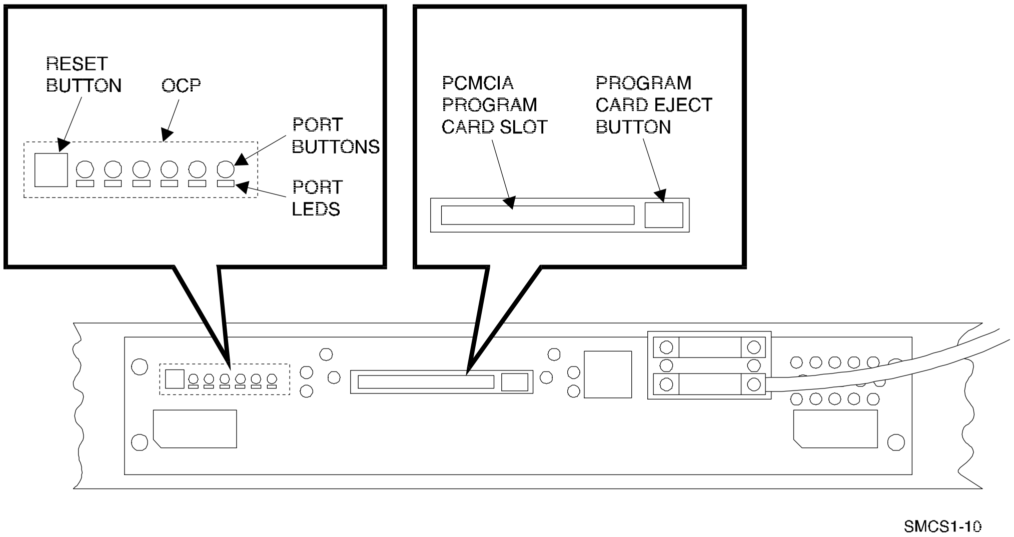
Overview

This chapter contains general information and a technical overview of the SWXRC-04 RAID Array controller.

1.1 Technical Overview

The SWXRC-04 RAID Array Controller provides high-performance, high-availability access to Small Computer System Interface 2 (SCSI-2) devices from a host computer system via the SCSI-2 interface. The SWXRC-04 controller provides a versatile, modular, solution to array storage problems and offers a wide variety of price and performance options. A convenient Operator Control Panel (OCP) on the SWXRC-04 controller (see Figure 1-1) provides a convenient method to control and monitor controller operation.

Figure 1–1 SWXRC-04 Controller and Operator Control Panel



1.1.1 Standard Features

The SWXRC-04 controller offers the following capabilities as standard features:

- Scaleable device capacity – Six separately-addressable SCSI device buses are supported. Up to 42 devices can be supported in a constrained single controller configuration.
- Device support – Device support is provided for rotating disks.
- **Logical Unit** Support – Up to eight logical units (LUNs) per controller **target ID** are supported.
- Multi-Target Support – Up to four target IDs per controller are supported.
- Intelligent write and **read cache** module (with **write-back** and **write-through** capability) and battery backup.
- **RAID** 0 support – Disk striping enhances performance by lowering latency and raising throughput. The SWXRC-04 controller's implementation of RAID 0 allows 2 to 14 members, with user-friendly **stripeset** configuration.
- Device **warm swapping** – The controller is seamlessly integrated with other StorageWorks components to enable device replacement while the system is operating, causing minimum impact on data integrity and system operation.
- Upgradeable controller firmware – A writeable Personal Computer Memory Card Industry Association **(PCMCIA)** flash memory card contains the controller's operating firmware and offers ease of installation and the ability to upgrade the controller firmware. The writeable firmware store also can be upgraded via a serial hardware port.
- Easy configuration of the storage subsystem – The controller and its physical storage devices are configured via a user-friendly command line interface **(CLI)**. The command line interface is accessed via a local maintenance terminal or with **virtual terminal** support from the host.
- Bad block replacement (BBR) and forced error (FE) support – The controller maintains data integrity by performing BBR activity in the controller, transparent to the host.
- Environmental Monitor Unit (**EMU**) support – The controller is used with a hardware EMU to enable it to monitor the physical condition of the storage subsystem.
- Code patching support – A resident utility program enables the program code in the controller's RAM to be manually modified.

1.1.2 Optional Features

The SWXRC-04 controller offers the following capabilities as options:

- Redundant controllers and controller caches – Tightly-coupled **dual-redundant** controllers (on the same host bus) offer automatic, intelligent, failover of attached storage in the event of a controller failure. Up to four targets and 36 SCSI devices can be supported by the two controllers in a dual-redundant configuration.

  The SWXRC-04 controller supports five different dual-redundant controller configurations:

  - Two controllers configured as a single, active controller with a hot spare backup. The active controller services from one to four targets.
  - Both controllers configured as active controllers, with one servicing one target, the other servicing one target.
  - Both controllers configured as active controllers, with one servicing one target, the other servicing two targets.
  - Both controllers configured as active controllers, with one servicing one target, the other servicing three targets.
  - Both controllers configured as active controllers, with one servicing two targets, the other servicing two targets.

  Failover actions within the storage subsystem are firmware-controlled and transparent to the host in all dual-redundant configurations.

- Controller and cache module warm swapping – In the dual-redundant configuration, failed controller and cache modules can be replaced without interrupting the operation of the subsystem.

1.1.3 Licensed Features

Licensed features are those that are designed into the controller firmware but must be enabled with special license keys via the user CLI. A license key is preentered into the SWXRC-04 controller firmware to enable the following licensed features:

- Intelligent write-back cache firmware – The write-back cache employs a hardware cache module with non-volatile storage that allows both read and write cache blocks to coexist in the cache. The ability to handle both read and write cache blocks simultaneously enables a controller with the write-back cache option to adapt to varying host I/O mixes without the typical loss in performance that fixed cache types experience. The cache implementation significantly reduces host I/O transfer latency in all configurations. Write-back caching must be enabled by the user via the CLI.
- RAID 5/3 storage firmware – The RAID (Redundant Array of Independent Disks) data structure uses the RAID 5 parity organization. When opportunities exist within a given subsystem, however, RAID 5 and RAID 3 techniques can be mixed to maximize performance and data integrity. Write aggregation, request combining, and cache flushing are implemented to enhance RAID performance. The RAID 5/3 storage firmware requires that the write-back module be installed to execute, even if write-back caching is not enabled by the user.

- RAID 1 storage firmware – RAID level 1 (mirroring) achieves data redundancy and increases availability by maintaining at least two drives that have exactly the same data. Thus, if one drive fails, another drive can provide an exact copy of the lost data immediately. Data is written to all drives in a mirrorset at once. The RAID 1 storage firmware requires that the write-back module be installed to execute, even if write-back caching is not enabled by the user.

1.1.4 Device Support

A current list of supported devices (see release notes) can be obtained from your service provider.

1.2 Controller Description

Figure 1-2 illustrates a block diagram of the SWXRC-04 controller.

The SWXRC-04 controller uses a single, fast, 16 bit, differential, SCSI bus for its connection to one or more host computers. The controller provides six, separately-addressable, single-ended, fast, 8-bit SCSI buses as its device interface. The SWXRC-04 supports all of the standard SCSI-2 features as well as many of the optional ones.

The SWXRC-04 controller can be configured alone or in conjunction with another controller for increased availability. The controller includes a 32 MB read/write-back cache module with battery backup.

The operator performs configuration of storage devices into logical units with interface firmware that is accessible via the SWXRC-04 maintenance port (an EIA-423 interface) or via a host-resident virtual terminal program.

When used in an enclosure that contains an environmental monitor unit, the SWXRC-04 controller detects fault signals from the EMU and alerts the operator in the event of potential storage subsystem damage or malfunction.

Controller hardware is designed to be installed in standard StorageWorks controller shelves and enclosures. Controller and cache modules are usually housed together in an **SWXSS-01 controller shelf** or an **SWXSC-AA storage enclosure**. These shelves can be inserted into various StorageWorks cabinets.

Figure 1–2 SWXRC-04 Controller Block Diagram

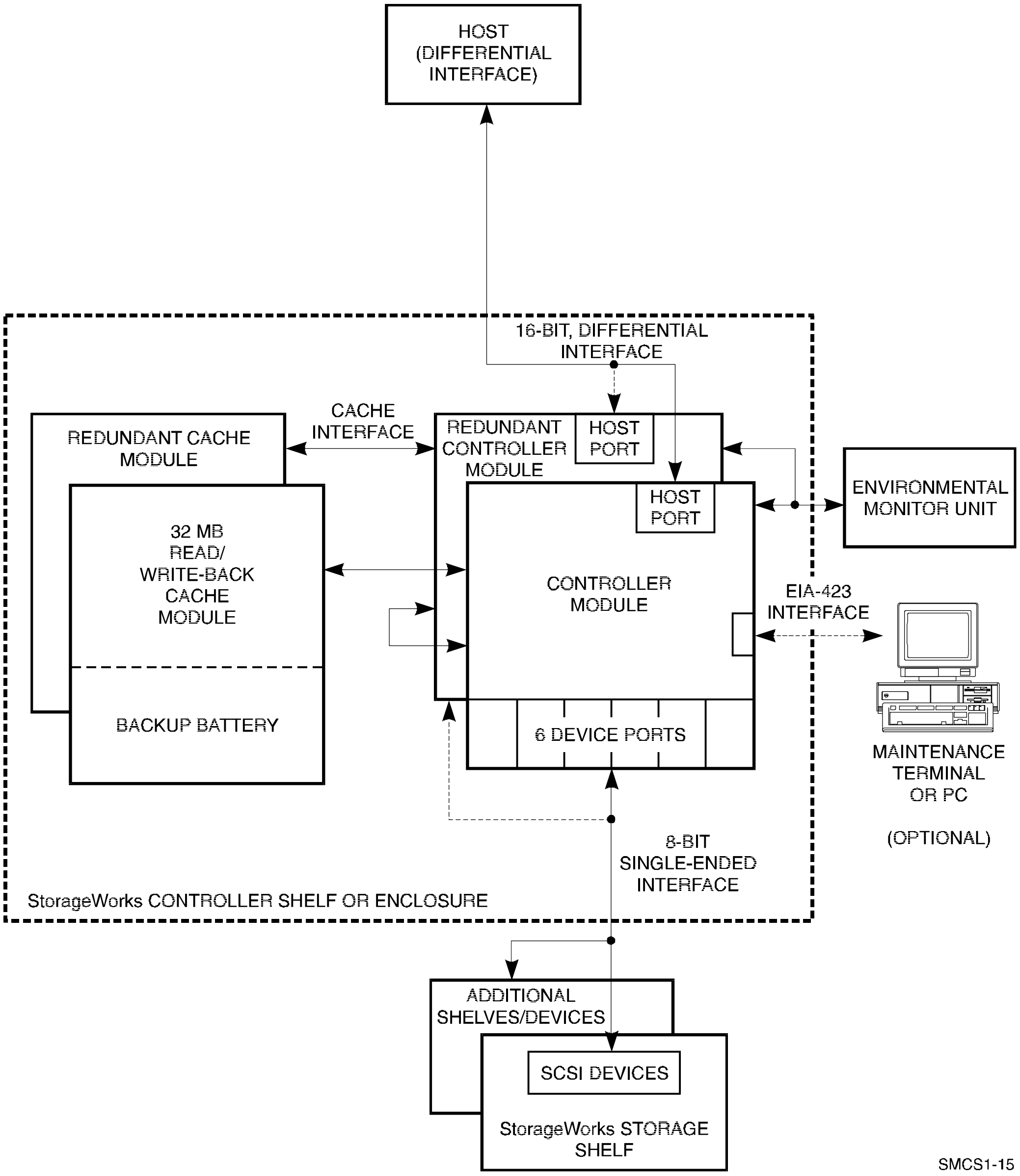


Table 1-1 summarizes the main features of the SWXRC-04 controller.

Table 1–1 Summary of SWXRC-04 Controller Product Features

Feature	SWXRC-04 Controller
Host system bus	SCSI-2, fast, wide, differential
Host protocol	SCSI-2
Storage device bus	SCSI-2, fast, narrow, single-ended
Storage device protocol	SCSI-2
Number of SCSI device ports	6
Number of SCSI-2 devices per port (SWXSS-06 SBB shelf)	6 (or 7)*
Number of SCSI-2 devices per port (SWXSC-AA storage enclosure)	4
Maximum number of SCSI-2 devices (SWXSS-06 SBB shelves)	36 (or 42)*
Maximum number of SCSI-2 devices (SWXSC-AA storage enclosure)	24
Dual-redundant configurations	Yes
Controller warm swap	Yes
Read/Write-back cache module	32 MB
Device warm swap	Yes
Controller-based device exercisers	Yes
Preferred ID reservation	Yes
Spontaneous messages to maintenance terminal	Yes
RAID level support	0/1/0+1/3/5
Program card firmware update	Yes (by replacing card or downloading the firmware)
Error detection code (EDC)	Validation of program card firmware
Error correction code (ECC) on cache and shared memory	Yes
Tagged command queuing	Yes
Power fail write nonvolatile journal	Yes
Data integrity and byte parity (all buses/memory)	Yes

* The dual-redundant controller configuration supports up to six devices per port. Nonredundant configurations support up to seven devices per port, but this sacrifices a convenient upgrade to high availability and redundant/backup power options.

1.3 Operating System Support

Refer to your firmware release notes (Solaris 1.1x, Solaris 2.x, HP-UX, IBM AIX) for restrictions and updates regarding operating system support.

1.4 Controller Module Specifications

Table 1-2 lists the physical and electrical specifications for the SWXRC-04 controller and its cache module. Measurements in Table 1-2 are nominal measurements; tolerances are not listed.

Table 1–2 Controller Module Specifications

Hardware	Length (mm/in)	Width (mm/in)	Power	Current at +5 V	Current at +12 V
Controller module	317.5/12.5	222.25/8.75	23.27 W	4.63 A	10 mA
32 MB Read/Write-back cache	317.5/12.5	196.85/7.75	2.48 W	400 mA	40 mA
module, (Battery Charging)			8.72 W	400 mA	560 mA

1.5 Controller Environmental Specifications

The SWXRC-04 controller is intended for installation in a Class A computer room environment. The StorageWorks product environmental specifications are listed in Table 1-3.

Table 1–3 Environmental Specifications

Condition	Specification	
	Optimum Operating Environment	
Temperature	+18° to +24°C (+65° to +75°F)	
	Rate of change	3°C (5.4°F) per hour
	Step change	3°C (5.4°F) per hour
Relative humidity	40% to 60% (noncondensing) with a step change of 10% or less (noncondensing)	
Altitude	From sea level to 2400 m (8000 ft)	
Air quality	Maximum particle count .5 micron or larger, not to exceed 500,000 particles per cubic ft of air	
Inlet air	.026 cubic m per second (50 cubic ft per volume minute)	

Table 1–3 Environmental Specifications (continued)

Condition	Specification
	Maximum Operating Environment (Range)
Temperature	+10° to +40°C (+50° to +104°F) Derate 1.8°C for each 1000 m (1.0°F for each 1000 ft) of altitude Maximum temperature gradient 11°C/hr (20°F/hr) ±2°C/hr (4°F/hr)
Relative humidity	10% to 90% (noncondensing) Maximum wet bulb temperature: 28°C (82°F) Minimum dew point: 2°C (36°F)
	Maximum Nonoperating Environment (Range)
Temperature	-40° to +66°C (-40° to +151°F) (During transportation and associated short-term storage)
Relative humidity Nonoperating	8% to 95% in original shipping container (noncondensing); otherwise, 50% (noncondensing)
Altitude	From -300 m (-1000 ft) to +3600 m (+12,000 ft), Mean sea level

1.6 Maintenance Strategy

The maintenance philosophy of the SWXRC-04 controller subsystem involves the removal and replacement of field replaceable units (FRUs) when failures occur. Chapter 9 contains FRU removal and replacement procedures. See Appendix A for a list of FRUs and FRU part numbers.

NOTE

Do not attempt to replace or repair components within FRUs. Use the controller internal diagnostics and error logs to isolate only to the FRU level.

1.7 Maintenance Features

The SWXRC-04 controller has the following features to aid in troubleshooting and maintenance:

- **Initialization diagnostics**

  Various levels of initialization diagnostics execute on the controller. These tests ensure that the subsystem is ready to come on line after it has been reset, powered on, and so forth. You can elect to rerun many of the diagnostics even after initialization completes, to test controller operation. See Chapter 8 for more information about controller initialization.

- **Utilities**

  You can run the VTDPY utility to display current controller state and performance data, including processor utilization, host port activity and status, device state, logical unit state, and cache and I/O performance.

  The configuration utility (CONFIG) checks the SCSI device ports for any device not previously added. This utility adds and names these devices.

  The Firmware Licensing System (FLS) enables and disables RAID, MIRRORING, and write-back caching, which are licensed controller features.

  The Fault Management Utility (FMU) controls some spontaneous error displays and displays controller last failure and memory system failure information.

  See Chapter 8 for detailed information on each utility.

- **Exercisers**

  The controller can run the disk inline exerciser (DILX). This exerciser simulates high levels of user activity, and running it provides performance information you can use to determine the health of the controller and devices attached to it. See Chapter 8 for more information about the exercisers.

- **Terminal access**

  You can use a virtual terminal (host terminal) or a maintenance terminal to check status and set operating parameters. The terminal connection provides access to the following:

  - CLI (See Chapter 5, Appendix B)
  - Error messages (See Chapter 7)
  - Error logs (See Chapter 7)

- **Controller warm swap**

  You can safely remove and replace, or warm swap, one controller in a dual-redundant configuration while the power is on. When you warm swap a controller, you are changing out a controller in the most transparent method available to the SWXRC-04 controller subsystem. Warm swapping a controller has minimal system and device impact. For more information on warm swapping, see Chapter 9.

- **Operator control panel**

  The operator control panel (OCP) on the front of the controller has seven buttons and LEDs. The buttons and LEDs serve different functions with respect to controlling the SCSI ports and/or reporting fault and normal conditions. See Chapter 7 for a complete description of the OCP.

1.8 Precautions

This section describes necessary precautions and procedures for properly maintaining and servicing SWXRC-04 controllers. Follow the guidelines in the following sections when performing any of the hardware maintenance actions outlined in this guide.

1.8.1 Electrostatic Discharge Protection

Electrostatic discharge (ESD) is a common problem for any electronic device and may cause data loss, system down time, and other problems. The most common source of static electricity is the movement of people in contact with carpets and clothing. Low humidity also increases the amount of static electricity. You must discharge all static electricity prior to touching the storage subsystem.

In general, you should follow routine ESD protection procedures when handling controller modules and cache modules and when working around the cabinet and shelf that houses the modules.

Follow these guidelines to further minimize ESD problems:

- Maintain more than 40-percent humidity in the room where the equipment is installed.
- Place the subsystem cabinet away from heavy traffic paths.
- Do not place the subsystem on carpet, if possible. If carpet is necessary, choose antistatic carpet. If the carpet is already in place, place antistatic mats around the subsystem.
- Use ESD wrist straps, antistatic bags, and grounded ESD mats when handling FRUs.
- Obey the module handling and grounding guidelines listed in Section 1.8.2.

1.8.2 Module Handling Guidelines

Prior to handling the controller module or cache module, follow these grounding guidelines. See Chapter 9 for module removal and replacement instructions.

- Obtain and wear an ESD wrist strap on your wrist. Make sure the strap fits snugly.
- Attach the lead on the ESD strap to a convenient cabinet grounding point.
- After removing a module from the shelf, place the module into an approved antistatic bag or onto a grounded antistatic mat.
- Remain grounded while installing a replacement module.

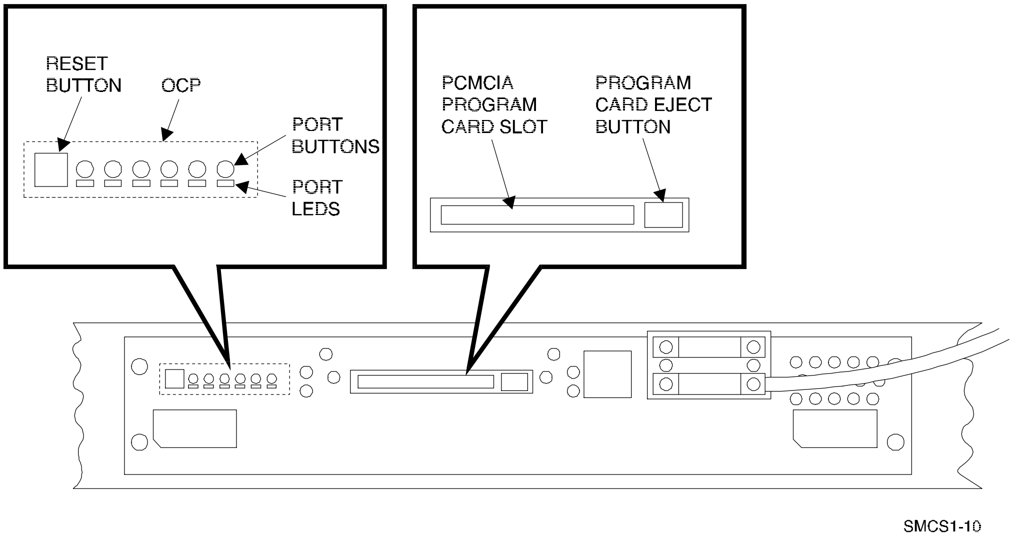
1.8.3 Program Card Handling Guidelines

Follow these guidelines when handling the program card. See Chapter 9 for program card removal and replacement instructions.

> **CAUTION**
>
> Follow program card guidelines or damage to the program card and firmware can result.

- Cover the program card with the snap-on ESD shield whenever the card is installed in the controller.
- Obtain and wear an ESD wrist strap on your wrist. Make sure the strap fits snugly.
- Attach your ESD strap to a suitable cabinet grounding point before removing, inserting, or handling the program card.
- Keep the program card in its original carrying case when not in use.
- Do not twist or bend the program card.
- Do not touch the card contacts.
- Keep the card out of direct sunlight.
- Do not immerse the card in water or chemicals.
- Always push the program card eject button, shown in Figure 1-3, to remove the card. Do not pull on the card.

Figure 1–3 Program Card Eject Button



2

Functional Description

This chapter provides a detailed functional description of the SWXRC-04 controller hardware and firmware.

2.1 Hardware

The controller provides a connection between a host computer and an array of SCSI-2 compatible storage devices. The controller hardware includes the following core circuitry:

- Policy processor
- Program card
- Diagnostic registers
- Operator control panel
- Maintenance terminal port
- Dual controller port
- Nonvolatile memory (NVMEM)
- Bus exchangers
- Shared memory
- Value-added hardware
- Device ports
- Cache module

Figure 2-1 shows a block diagram of the SWXRC-04 controller hardware.

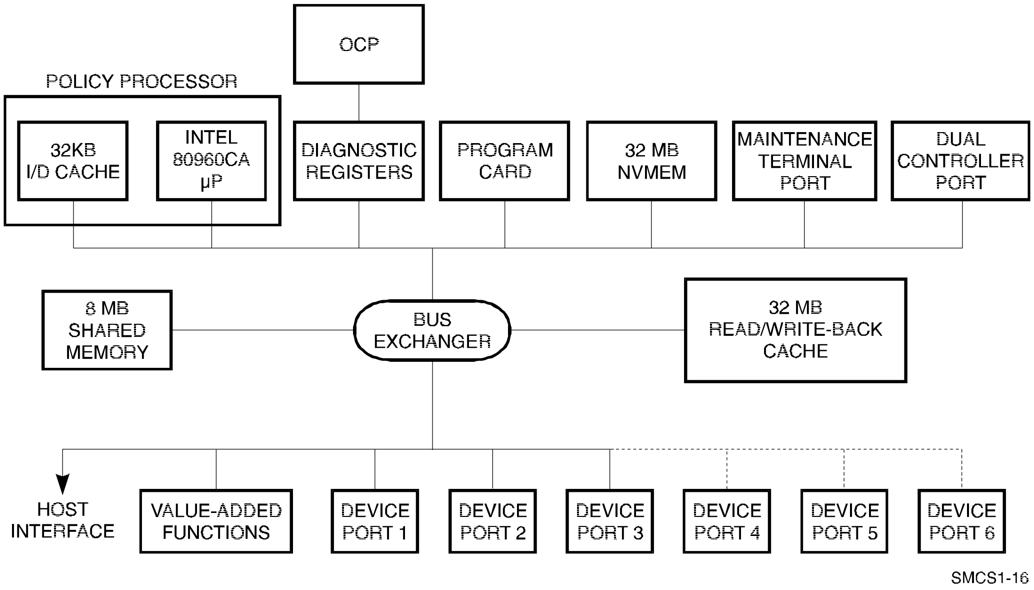
2.1.1 Policy Processor

The policy processor is the microprocessor hardware necessary for running the controller's firmware.

2.1.1.1 Intel 80960CA

The heart of the policy processor is an Intel® 80960CA (i960) processor chip. This processor chip runs the firmware from the program card and provides a consistent 25 MIPs. The i960 controls all but low-level device and host port operations.

Figure 2–1 Controller Hardware Block Diagram



2.1.1.2 Instruction/Data Cache

Although the i960 has an internal cache, the internal cache is not large enough to offset performance degradation caused by shared memory. To compensate for this, the i960 utilizes a separate instruction/data (I/D) cache. This 32-KB static RAM (SRAM) cache helps the i960 achieve faster access to instructions and variables. A **write-through cache** design maintains data coherency in the I/D cache.

2.1.2 Program Card

The program card is a PCMCIA standard flash card device containing the firmware for operating the controller. The firmware is validated and then loaded from the program card into shared memory each time the controller initializes. Special error correction code corrects program card errors when possible. The program card can be rewritten, providing a convenient method of updating the controller's firmware.

2.1.3 Diagnostic Registers

The controller has two write and two read diagnostic registers. Diagnostic and functional firmware use the write diagnostic registers to manipulate controller and device operations. Certain bits in the registers activate test modes for forcing errors in the controller. Other bits control the operator control panel (OCP) LEDs. The policy processor reads the read diagnostic registers to determine the cause of an interrupt, when an interrupt occurs.

2.1.4 Operator Control Panel

The OCP includes the following:

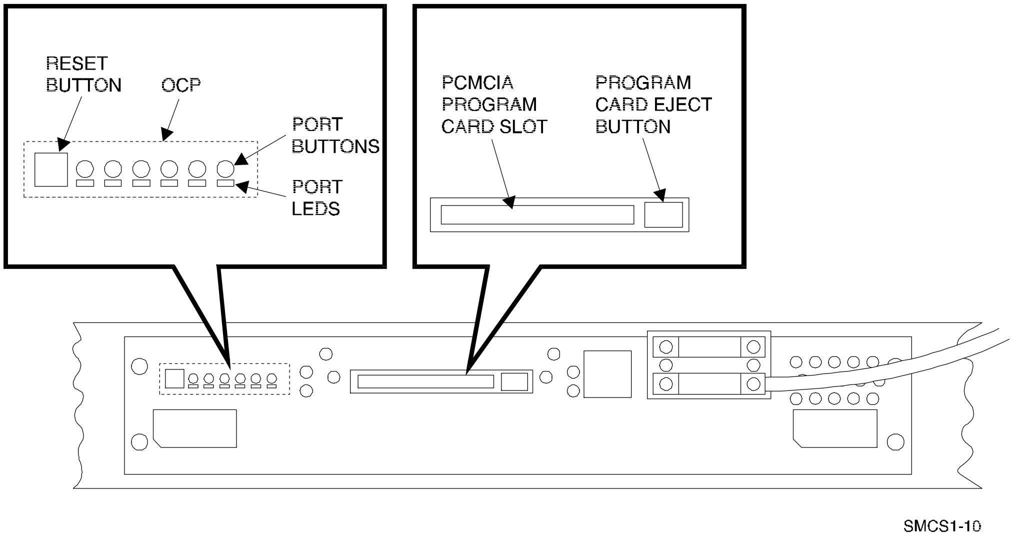
- One reset button with embedded green LED
- One button per SCSI port
- Six amber LEDs

Figure 2-2 shows the SWXRC-04 controller OCP. The buttons and LEDs serve different functions with respect to controlling the SCSI ports and/or reporting fault and normal conditions:

- The green reset LED indicates normal operation and fault conditions.
- The amber port LEDs indicate errors and faults.
- The reset button halts all activity on the controller and causes it to reboot.
- Each of the port buttons quiesces (pauses) the associated device bus.

See Chapter 5 for further information on using the OCP.

Figure 2–2 Controller Operator Control Panel



2.1.5 Maintenance Terminal Port

Each SWXRC-04 controller has a modified modular jack (MMJ) on its front bezel that can support an EIA-423 compatible maintenance terminal. You must connect a maintenance terminal during controller installation to set initial controller parameters. During normal operation, you may use either a maintenance terminal or a virtual (host) terminal to add devices and **storagesets**, or to perform other storage configuration tasks. However, a maintenance terminal is required when a host connection is not available.

NOTE

If you connect a maintenance terminal to one controller in a dual-redundant configuration and both controllers are functioning, you can communicate with both controllers through this connection.

2.1.6 Dual Controller Port

The controller has an internal serial port for communication with a second controller of the same model. The second controller needs to be mounted in the same controller shelf, with communication passing through the ports and across the shelf backplane. A dual-redundant configuration enables one controller to take over for another (failed) controller. The takeover process is called **failover**. During failover, the **surviving controller** supports the SCSI-2 devices linked to the failed controller. See Chapter 5 for more information on failover.

2.1.7 Nonvolatile Memory

The controller has 32 KB of nonvolatile memory **(NVMEM)**. NVMEM is implemented using battery backed up SRAM. This memory serves two purposes:

- NVMEM stores parameter and configuration information such as device and **unit** number assignments entered by the user and by the controller firmware.
- NVMEM stores cache module metadata, which enables the controller to check for the correct cache module.

2.1.8 Bus Exchangers

Bus exchange devices enable high-speed communication between bus devices and shared memory. One bus exchanger handles address lines while the other exchanger handles data lines. The bus exchangers are classified as four-way cross-point switches, which means the bus exchangers enable connections between one port and any other port on the switch.

2.1.9 Shared Memory

Shared memory consists of a dynamic RAM controller and arbitration engine (DRAB) gate array controller and 8 MB of associated dynamic RAM (DRAM). Shared memory uses parity-protected 9-bit error correction code (**ECC**) and error detection code (**EDC**) for improved error detection and data recovery. The shared memory also stores the controller firmware and is shared between bus devices for data structures as well as data buffers.

One portion of shared memory contains instructions for the i960 chip, firmware variables, and data structures, including the look-up table for the i960 chip. In the absence of the cache module, another portion of shared memory acts as a cache. Otherwise, this portion contains cache module context for cache look-ups when a cache module is in place.

2.1.10 Value-Added Functions

This circuit acts as the accelerator for RAID XOR operations as well as device compare operations. To support recovery of a lost member of a RAID5/3 storageset, the controller will perform an XOR of parity data with the surviving **RAIDset** members' data to **reconstruct** the lost member.

2.1.11 Device Ports

The controller SCSI-2 device ports are a combination of NCR® 53C710 SCSI port processors and SCSI transceivers. The 53C710 processors perform operations in 8-bit, single-ended normal or fast mode. The 53C710 processors execute scripts read from shared memory and under control of the policy processor.

Each SCSI-2 port can have up to six or seven attached devices depending on controller configuration (dual-redundant or nonredundant, respectively). In a dual-redundant configuration, device availability improves because each controller has access to the other controller's devices.

2.1.12 Cache Module

The SWXRC-04 controller is enhanced with a companion 32 MB **read/write-back cache** module.

2.1.12.1 Common Cache Functions

The cache module increases controller I/O performance. During normal operation, a host read operation accesses data either from the fast memory of the cache module or from an I/O device.

If a host read is a cache “hit” (data already in the cache), the data is supplied to the host immediately, improving I/O performance by reducing latency. If the host read is a cache “miss” (data not in the cache), the controller accesses the appropriate device to satisfy the request. Then the controller reads the data, returns it to the host, and writes it to the cache.

Cache entry sizes are fixed at 64 KB (128 logical **blocks**) for each logical unit. Read caching is enabled by default but can be optionally disabled using the CLI> SET command on a per unit basis (see Appendix B.)

The data replacement algorithm is a least recently used (**LRU**) replacement algorithm. When the cache is full and new data must be written, the LRU algorithm removes the oldest resident cached data with the least number of references and replaces it with the new data.

2.1.12.2 Read Cache Operation

During a host write operation with the **read cache** enabled, data is written to the disk *and* the cache. This is known as write-through caching, and it improves the performance of subsequent reads, because often the requested data was previously written to the cache.

The read cache consists of DRAM storage. However, the read cache is volatile. Subsystem power failures cause the loss of all data in the read cache.

2.1.12.3 Write-Back Cache Operation

The write-back cache operation increases subsystem performance as well as preserves data integrity under power failure situations.

In write-back caching, data is not always written to storage and cache simultaneously (write-through caching). Instead, data intended for storage may remain in the cache until the optimum time to write, or **flush**, to a device occurs. When data is suspended in this way, it is referred to as **unwritten cache data**. A power failure in conjunction with unwritten cache data has disastrous consequences because the information is lost. In RAIDset configurations, the impact of power failure is worsened by the possibility of **write hole** data loss as well.

For this reason, the write-back cache utilizes onboard rechargeable batteries. The batteries power the memory to retain data when cache power is intentionally or accidentally interrupted. The battery circuit automatically detects loss of power and switches from shelf backplane power to battery power.

Battery Discharging

During a power failure, the write-back cache batteries discharge very slowly, providing standby power to retain cache data. Under these circumstances, Digital guarantees data retention time for fully charged batteries to be a minimum of 100 hours.

Studies performed by utility companies have noted that over 99 percent of all power outages last less than one minute. However, once power fails for more than 3 minutes, the controller modifies its operation upon restart, to maximize data preservation:

- Stripesets and disk-based units with write-back caching enabled are accessed in write-through (read cache) mode, until the cache batteries are fully recharged. Once the batteries are recharged, write-back caching resumes.
- RAIDsets and Mirrorsets will be available either in write-through mode, or unavailable until the batteries are fully recharged from a long power failure, depending upon the cache policy set for the controller. See Chapter 7 for a full explanation of write-back cache battery policy. RAIDsets do not failover to the other controller in the event of a battery failure.

Battery Charging

Under normal conditions (power restored), a battery charge circuit senses the battery voltage and automatically activates a charger to achieve and maintain full battery charge. The following ratio illustrates the relationship of power down time versus battery recharge time:

32 MB write-back cache-12:1 (power down time v. time to full charge)

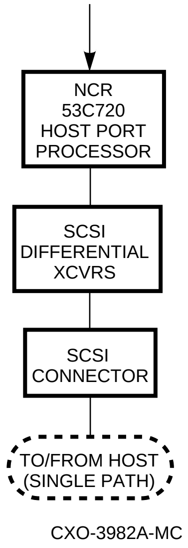
For example, a two hour power outage results in approximately 10 minutes of recharge time for a 32 MB write-back cache. The recharge time is a safety precaution, taken to preserve the integrity of the controller subsystem for any power outages that may occur thereafter.

> **NOTE**
>
> Digital recommends replacing the write-back cache batteries at five (5) year intervals.

2.1.13 Host Interface

Figure 2-3 shows a block diagram of the SWXRC-04 to SCSI-2 host interface hardware.

Figure 2–3 SCSI–2 Host Interface Hardware Block Diagram



The SWXRC-04 interfaces with a fast, wide, differential SCSI (FWD SCSI) 16-bit host bus. The hardware consists of the NCR 53C720 chip and transceivers.

2.2 Firmware

The controller operating firmware consists of functional code, diagnostics, utilities, and exercisers.

The controller operating firmware is stored in a writeable PCMCIA program flash card. Digital ships the card along with your controller. Thereafter, each time SWXRC-04 operating firmware is updated, new cards are manufactured. You can also download new versions of the firmware directly into your program card.

Once the card is installed in the controller, the contents are validated and loaded into shared memory. Special error correction code corrects program card errors when possible. Any time you reset the controller, this validating and loading process gets repeated. Because of this scheme, when the firmware executes, only part of the controller initialization diagnostics runs directly from the program card. The remaining diagnostics, all functional code, and all utilities run from controller shared memory.

The controller operating firmware consists of five functional areas:

- Core functions
- Host interconnect functions
- Operator interface and subsystem management functions
- Device services functions
- Value-added functions

These functions are discussed in the following sections.

2.2.1 Core Functions

SWXRC-04 operating firmware provides the following core functions, in the order they are executed following turning on the controller:

1. Tests and diagnostics
2. Executive functions

2.2.1.1 Tests and Diagnostics

Controller tests and diagnostics are integrated in a controller self-test procedure performed when the controller initializes. The output of the self-test procedure is a simple go/no go status of the controller subsystem. The self-test procedure includes a test of the cache module. See Chapter 8 for additional initialization and self-test information.

2.2.1.2 Init Functions

After successful completion of the tests and diagnostics, the controller data structures are initialized and memory is allocated for controller functions.

2.2.1.3 Executive Functions

Firmware executive (EXEC) functions act as the operating system kernel for the SWXRC-04 controller. EXEC functions control firmware execution with respect to interrupts, thread control, queuing support, timers, and so forth. The EXEC functions establish the controller environment as a non-preemptive interrupt-driven process.

2.2.2 Host Interconnect Functions

The SWXRC-04 controller host interface uses the SCSI-2 protocol with SCSI pass through software to the CLI, **tagged command queuing**, and mode select/sense support for SCSI.

2.2.3 Operator Interface and Subsystem Management Functions

The operator interface and subsystem management functions support the user interface, subsystem management, subsystem verification, and error logging/fault management. These functions are presented in the following sections.

2.2.3.1 Command Line Interpreter

The Command Line Interpreter (CLI) is the primary user interface for the controller. The CLI contains firmware for responding to most management functions plus local program execution. Briefly, the CLI provides the following two types of commands:

- SET/SHOW commands for the controller itself. This includes setting and showing the controller ID, name, path controls, and other vital information.
- Configuration commands to add/delete devices, storagesets, and logical units.

See Appendix B for more information on the CLI and its commands.

2.2.3.2 Local Programs

There are several local exercisers and utilities available for controller subsystem management/verification, as follow:

DILX is a disk exerciser that enables you to test and verify operation of the controller with attached SCSI-2 storage under a high or low I/O load. This utility places the load on the controller, bypassing the host port. Chapter 8 provides a full description of DILX.

VTDPY enables you to display current controller state and performance data, including processor utilization, host port activity and status, device state, logical unit state, and cache and I/O performance. See Chapter 8 for detailed information on this utility.

Controller warm swap (C_SWAP) enables you to safely remove and replace one controller in a dual-redundant configuration while the system is operational. When you warm swap a controller, you are replacing a controller in the most transparent method available to the controller subsystem. Warm swapping a controller has minimal system and device impact, as explained in Chapter 9.

Firmware licensing system (FLS) enables a customer or field service engineer to observe and control licensed features within SWXRC-04 operating firmware. Any feature turned on without entering a valid license code (key) causes a message to be displayed on a connected maintenance terminal every hour. After entering a valid license key, these notifications stop. See Chapter 8 for more information on FLS.

Configuration menu (CFMENU) enables you to quickly configure storage devices attached to the controller. CFMENU presents configuration commands normally entered at the CLI in a menu-driven format. See Chapter 8 for more information on CFMENU.

Code Load/Code Patch (CLCP) Downloads new firmware code into the controller and enables changes or repairs to the controller's firmware without installing a new program card. Downloaded firmware and patches become active once you restart the controller. See Chapter 8 for more information on CLCP.

Fault Management Utility (FMU) controls display of some spontaneous event, controller last failure, and memory system failure information. FMU can also be used to view recent, last failure, and event messages. See Chapter 8 for more information on FMU.

Configure (CONFIG) checks the SCSI device ports for any device not previously added. This utility adds and names these devices. See Chapter 8 for more information on the configuration utility.

Clone Utility (Clone) creates a duplicate of any stripeset, mirrorset, or single-disk unit. You can use this utility to perform a backup of a unit (for example, a stripeset) without significantly impacting performance for users by first cloning the unit and then performing backup from the clone. See Chapter 8 for more information on the clone utility.

2.2.3.3 Event Reporting and Fault Management

Event Reporting and Fault Management is an integrated function that controls reporting of significant events detected during controller operation. Fault Management firmware recommends repair actions when faults are detected. See Chapter 7 for more information on these subjects.

2.2.4 Device Services

SCSI-2 device service firmware includes device port drivers and physical device addressing and access. Device service consists of normal functions such as read, write, and error recovery code. It also contains firmware for controlling and observing StorageWorks building blocks (**SBB**s) and shelves, such as monitoring LEDs, power, and **blowers.** Specific features include the following:

- Normal SCSI-2, 8-bit, single-ended support.
- FAST, synchronous, 8-bit, single-ended device support.
- Tagged queueing for SCSI-2 devices.
- Read and write physical device addressing and access. This is the read and write path to and from devices, and from and to the value-added portion of SWXRC-04 operating firmware.
- Specified device support per SWXRC-04 operating firmware release. Refer to your SWXRC-04 operating firmware release notes to identify specifically supported devices.
- Device warm swap. You can remove and replace most devices without taking the subsystem off line (see Chapter 9). See your firmware release notes for any restrictions on warm swapping devices.
- **Device shelf** and SBB observation and control. This service monitors SHELF_OK signals and alerts you of blower and power supply failures. This firmware also controls the fault LEDs on the SBBs for use in warm swap and identifying device failures or configuration mismatches.
- Device error recovery. This service performs error recovery and read and write retries directly, making every attempt to serve data to and from the host before declaring an unrecoverable error or marking a device as failed.
- Controller warm swap. This feature works under control from a local program running from the CLI. The program must quiesce all the SCSI buses to safely enable information).

2.2.5 Value-Added Functions

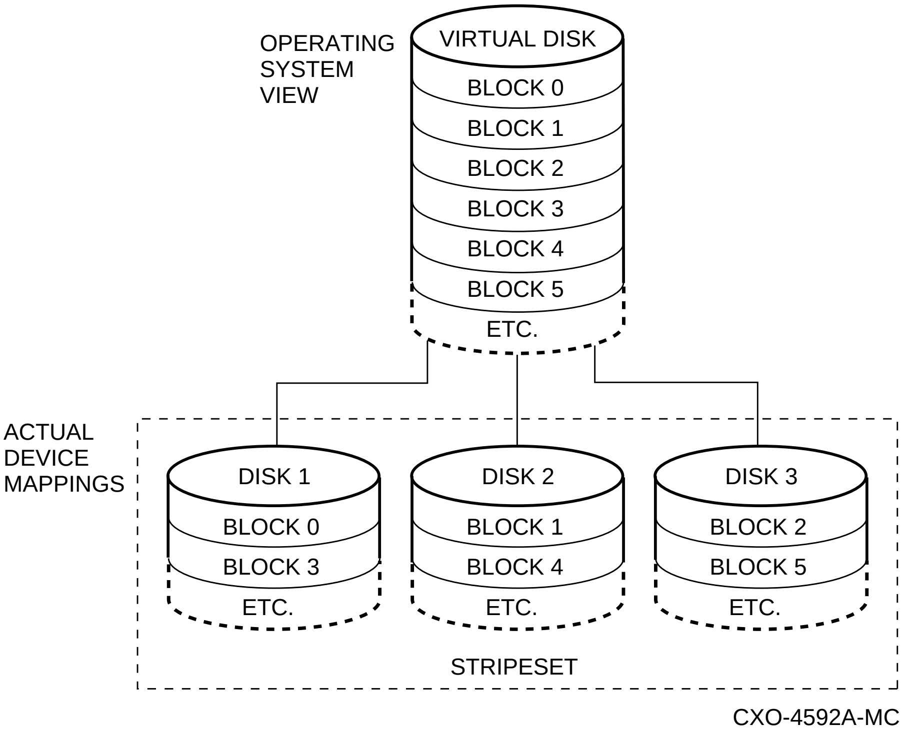
SWXRC-04 operating firmware contains value-added functions to enhance availability, performance, subsystem management and maintenance, and connectivity features of the controller subsystem. These value-added functions are presented in the following sections.

2.2.5.1 RAID

SWXRC-04 operating firmware supports levels of Redundant Array of Independent Disks (RAID) storage methods as follows:

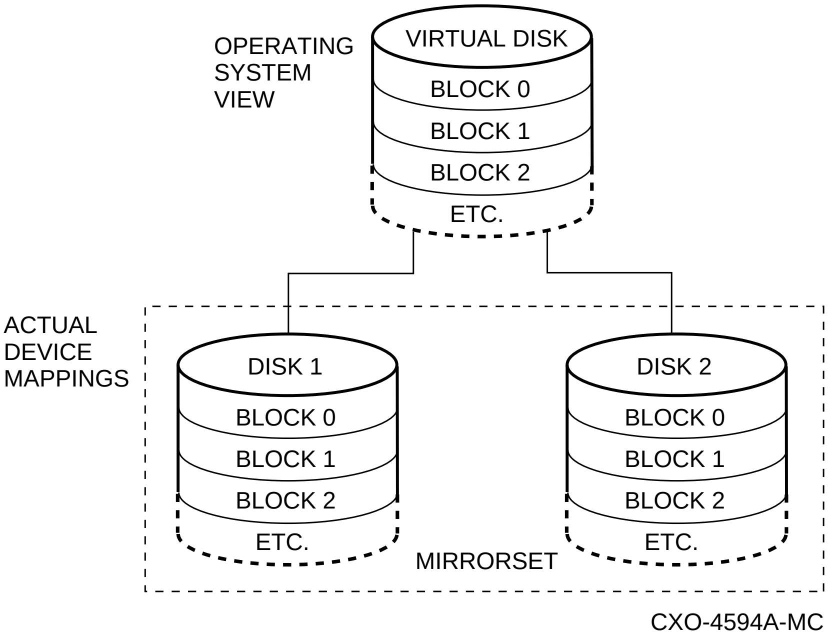
- RAID level 0 (striping). Striping spreads user data across multiple drives, as shown in Figure 2-4. Because reads and writes are performed in parallel to all **stripeset** members, this feature enhances performance in the areas of latency and throughput. Stripesets can be from 2 to 14 members. Striping firmware is tuned to balance the load across devices and not form maximum data transfer bandwidth.

Figure 2–4 RAID 0 – Striping



- RAID level 1 (mirroring). Mirroring achieves data redundancy and increases availability by maintaining at least two drives that have exactly the same data, as shown in Figure 2-5. Thus, if one drive fails, another drive can provide an exact copy of the last data immediately. Data is written to all drives in a mirrorset at once. Mirrorsets can have from 1 to 6 members.

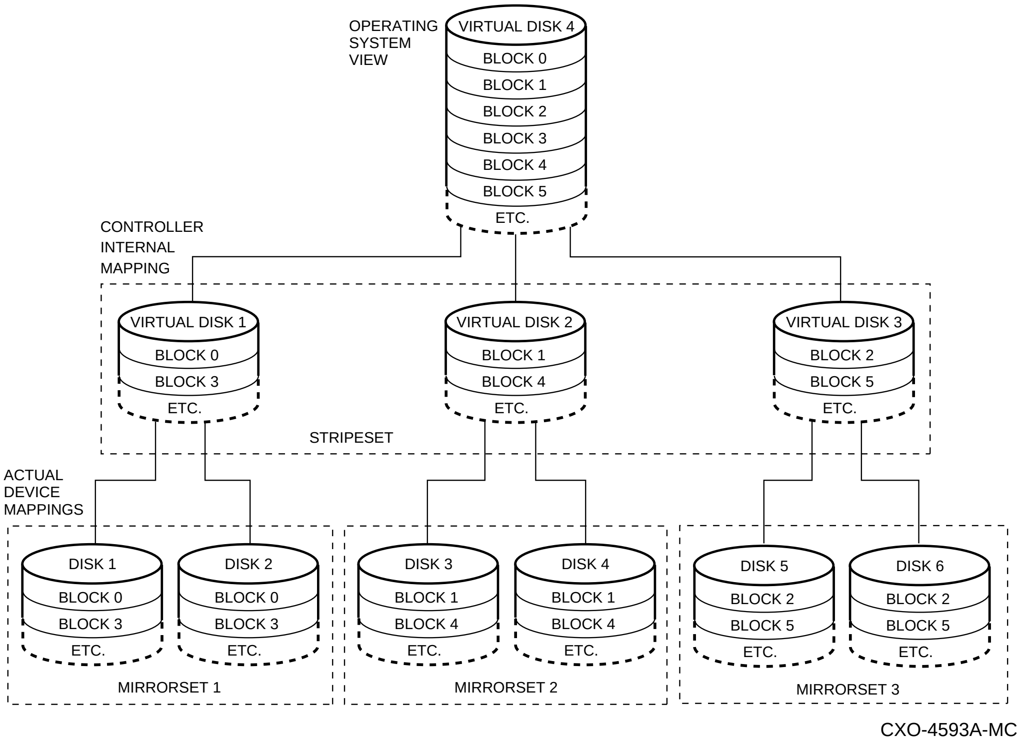
Figure 2–5 RAID 1 – Mirroring



Mirroring requires the write-back cache module option to ensure data integrity. Both the write-back cache and mirroring are licensed firmware features.

By combining RAID level 0 and RAID level 1, you can create a striped mirrorset, in which each member of a stripeset is actually a mirrorset. This configuration is shown in Figure 2-6, which can provide both throughput and availability.

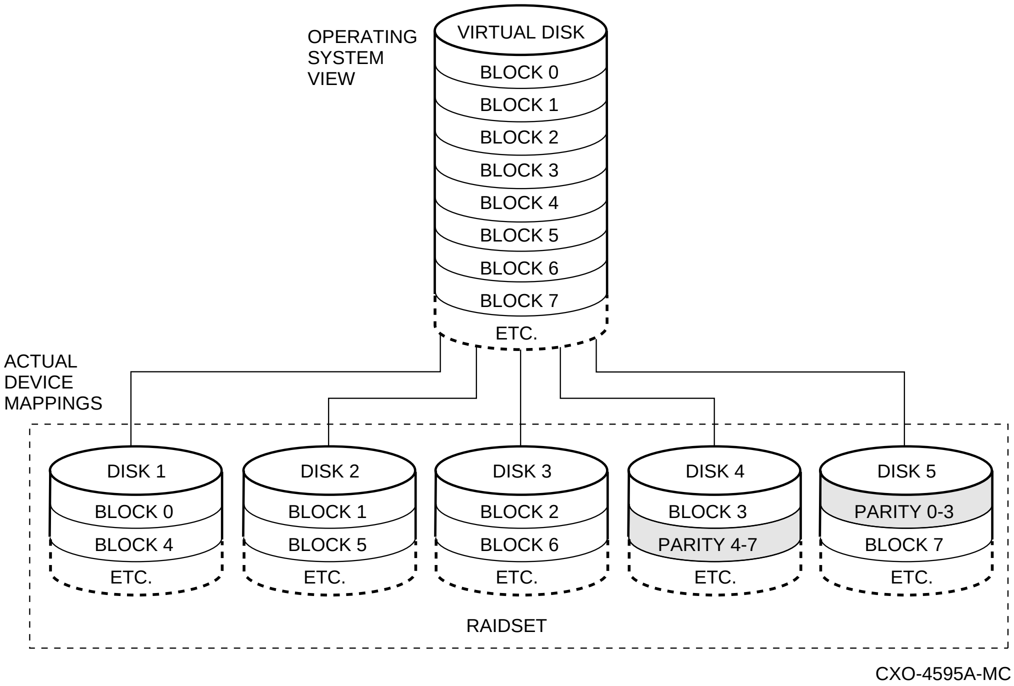
Figure 2–6 Striped Mirrorset



Mirroring is performed by the controller below the visibility of the host computer. Just as a host computer is not aware that a logical unit is actually a stripeset (as opposed to a single device), it also is not aware that a unit may be mirrored.

- RAID level 5. RAID 5 provides increased availability and throughput (as compared to a single disk or an unassociated group of disks) by striping data across a set of drives, and calculating and storing the parity information for each stripe. A RAIDset uses the capacity of one member of the set to store the parity information, but writes the parity blocks to different disks for each successive data stripe, as shown in Figure 2-7.

Figure 2–7 RAID 5 – Parity RAID



RAID 5 under the controller operating firmware includes the following features:

- Fast initialization
- RAIDsets with 3 to 14 members
- Nonredundant write, read/modify/write, reconstruct/write
- Read, reconstruct read, repair
- Forced error promotion, which allows for redundancy even when a block has been marked with forced error
- Automatic removal of a member based on error history
- Automatic member **replacement** from spare disks
- Reduced operation of RAIDsets that are missing one member
- Reconstruct scanning, to restore a RAIDset to a consistent state

RAID 5 requires the write-back cache module option to ensure data integrity. Both the write-back cache and RAID 5 are licensed firmware features.

Controller operating firmware also supports RAID level 3, because RAIDsets created with the controller operating firmware can achieve the high-performance characteristics of RAID 3, depending on how I/O transfers are tailored:

- Large I/O transfers (or many small transfers executed sequentially) will result in high bandwidth, with RAID 3 performance.
- Setting the RAIDset chunksize (see the INITIALIZE command in Appendix B) to a smaller value in conjunction with the large I/O transfers also will result in better RAID 3 performance.
- You must be enable RAIDset write-back caching to achieve RAID 3 performance.
- The speed of your host interface can impact the degree of RAID 3 performance you will achieve.

Note that by varying RAIDset chunksize, you can conveniently choose between more bandwidth-oriented or more throughput-oriented performance under the same RAIDset. Furthermore, by specifying an intermediate chunksize, you realize a combination of benefits – RAID 3 technology (for very large I/O operations) and RAID 5 technology (for small I/O operations).

Refer to the Digital Guide for RAID Storage Technology for a description of RAID and how the various levels of RAID improve data integrity and error recovery.

2.2.5.2 Failover

A failover component (FOC) in SWXRC-04 operating firmware links two controllers in a dual-redundant configuration. The controllers exchange status signals and configuration information. When one controller fails, the surviving controller takes over service to the failed controller's units. FOC communication enables for easier system management, because only one terminal connection is required to access both controllers. See Chapter 2, Section 2.4, and Chapter 5 for more information on failover.

2.2.5.3 Caching

Cache firmware within the value-added section of SWXRC-04 operating firmware addresses the following areas:

- Read caching
- Write-through caching
- Write-back caching for the battery backed up cache module. Write-back caching provides low write latency.
- Managing of 32 MB of cache
- RAID assistance for improved performance
- Logical Block Number (LBN) extent locking

- Least Recently Used (LRU) replacement policy (Refer to Section 2.1.12.1 for a description of the LRU algorithm.)
- Write-back cache module failover
- Caching enabled on a per unit basis

The cache policies for the product are as follow:

- Transfer defined extent (TDE) based cache.
- Transfer size-based data caching; maximum read/write size is changed on a per unit basis.
- I/O is subject to locking.

2.3 Addressing Storage Within the Subsystem

This section provides an overview about how storage is addressed in a controller subsystem. Storage is seen in two different ways, depending on your perspective:

- From the controller SCSI device interface – At the physical device level
- From the host interface – At the virtual device level

Following are descriptions of both levels of storage addressing.

2.3.1 Controller Storage Addressing

Figure 2-8 shows a typical physical storage device interface for a controller. Each of the controller's six device ports supports a SCSI bus connected with up to six devices. The devices typically reside in a StorageWorks SWXSS-02 SBB shelf. The current implementation of SWXRC or SWXSC-AA-series controllers supports only one controller LUN per physical device. LUN 0 is the default controller LUN address for each device.

Controller Port/Target/LUN Addressing

Controller port/target/LUN (PTL) addressing is the process by which the controller selects storage space within a specific, physical, storage device. The process takes place in three steps:

1. The **port** selection – The controller selects the SCSI bus port connected to a particular device.
2. The **target** selection – The controller selects the device's SCSI ID (that is, the target) on that port.
3. The **LUN** selection – The controller selects the desired LUN within that physical device. (In the current implementation, there is only one LUN on each device, and its LUN address is always 0.)

Note that controller PTL addressing is always tied to a physical storage device.

2.3.2 Host Storage Addressing

A typical host device interface consists of a number of host ports each connected to a bus containing devices. From the host's perspective, the controller is one of these devices.

To support certain high-level storage subsystem functions such as RAID, the controller presents the entire physical device configuration (from Figure 2-8) to the host as host logical units. A host logical unit can consist of a single device of a storageset comprised of more than one physical device. For this reason, a host logical unit is often called a virtual device.

You configure host logical units using the CLI.

Figure 2–8 Controller Storage Addressing

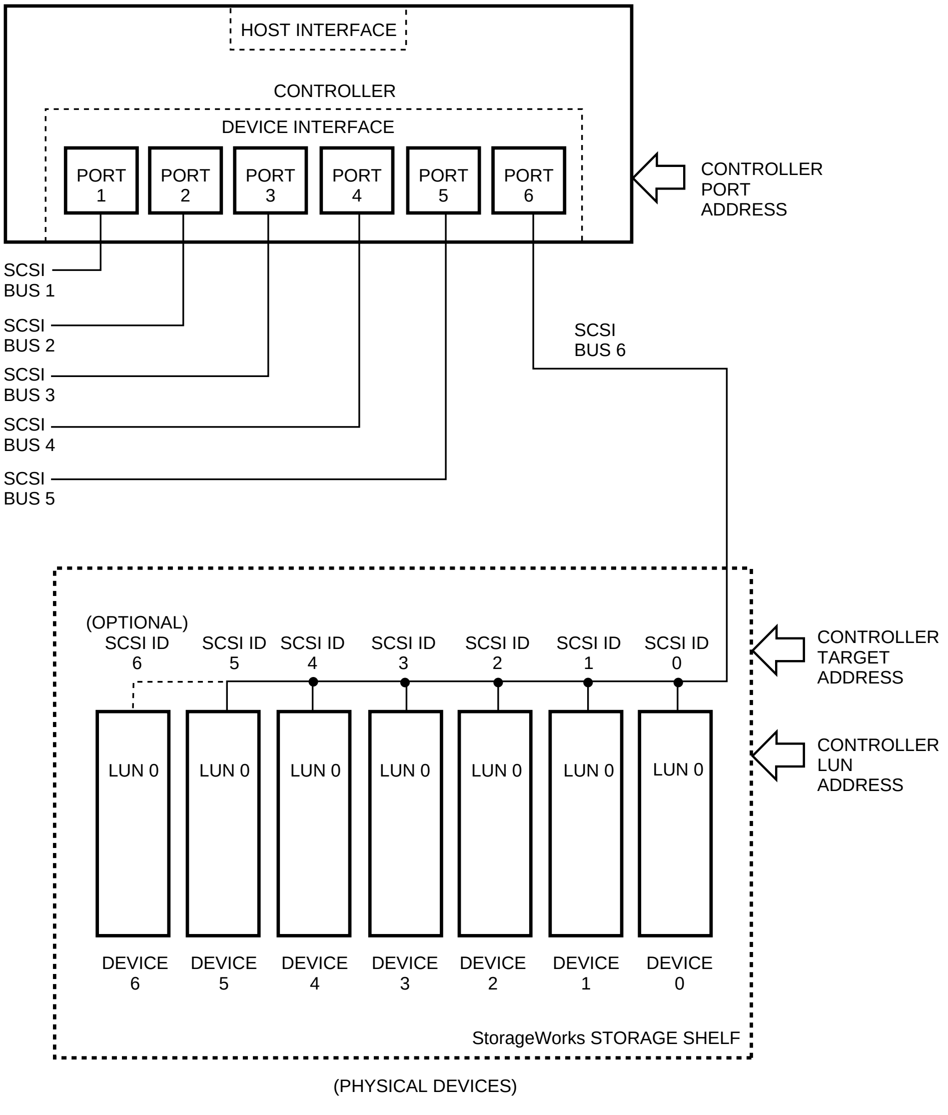
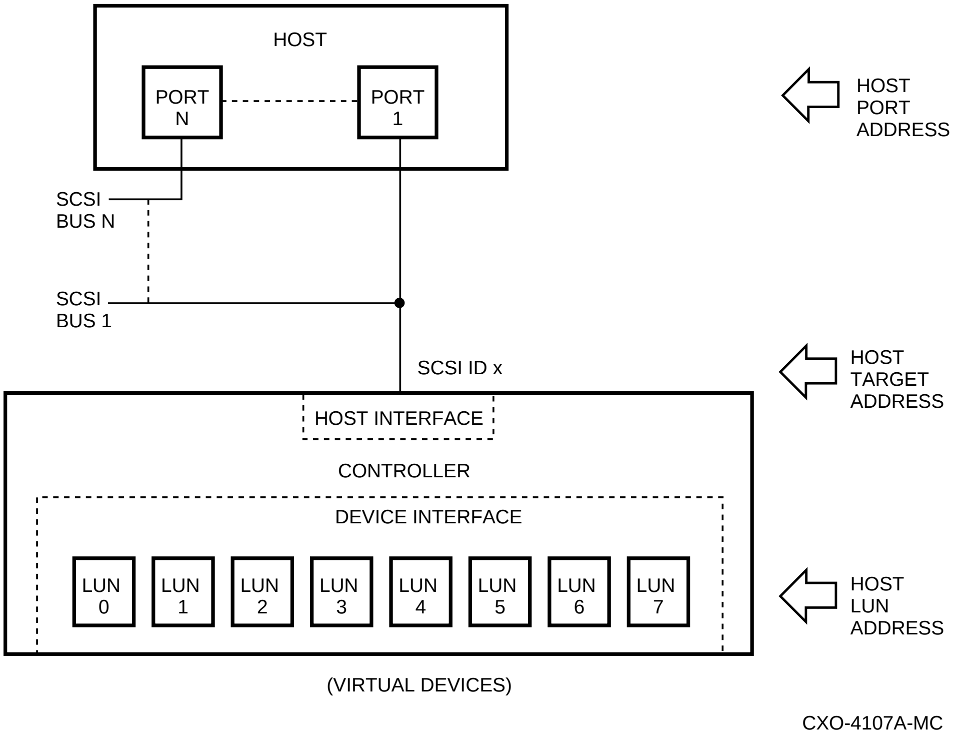


Figure 2-9 shows a typical connection between an SWXRC- 04 controller and its host. In this case, the SCSI host device interface consists of device ports, each connected to a SCSI bus containing up to eight devices. The SWXRC-04 controller resides on one of the SCSI buses.

The SWXRC-04 controller can be assigned from one to four SCSI IDs on the bus. This allows the controller to present more LUNs to the host, because each SCSI ID can only support eight LUNs. A controller with four SCSI IDs can present a maximum of 32 LUNs to the host.

Figure 2–9 Host Storage Addressing



NOTES

Even though the SWXRC-04 is a wide SCSI device, all devices on its SCSI bus must use SCSI IDs of 7 or less. Otherwise, the system will not operate correctly.

Although they share the same name, controller LUNs and SCSI host LUNs are logical addresses for two different storage structures. Controller LUNs exist on the *controller's* device interface, and SCSI host LUNs exist on a SCSI *host's* device interface.

Controller LUNs and SCSI host LUNs *may* represent the same structure, but only if the user configures controller devices in a one-to-one unit relationship with the host. This situation could occur under normal operation.

Host Port/Target/LUN Addressing

Host PTL addressing is the process by which a SCSI host selects a logical unit comprised of physical devices connected to an SWXRC-04 controller. The process takes place in three steps:

1. The port selection – The host selects the SCSI bus that has the SWXRC-04 controller connected to it.
2. The target selection – The host selects the controller's SCSI ID (that is, the target) on that port/bus. The SWXRC-04 controller may represent from one to four target IDs.
3. The LUN selection – The host presents the controller with the LUN of the desired host logical unit. The controller translates the LUN into the physical device addresses required to enable the host access to the virtual device.

3

Installing the Storage Subsystem

Digital offers preconfigured controller subsystems and controller subsystems configured-to-order (CTO) to your specific needs. You also can order individual storage subsystem components and assemble your own custom subsystem. This chapter presents the information necessary to perform the site installation of any of these SWXRC-04 controller storage subsystems.

3.1 Before You Begin Your Installation...

Before you begin installing your SWXRC-04 controller subsystem, consider the following:

- How many people are needed for unpacking and installation?
- What type of tools are needed for unpacking and installation?
- What ESD protection is required?
- What precautions should you be aware of in handling subsystem components?
- Have you prepared your site for the installation?

The following sections discuss these considerations.

3.1.1 Personnel Needed for Installation

The number of people needed to install an SWXRC-04 controller subsystem depends on the size and weight of the subsystem cabinet. For more information, refer to the appropriate *Getting Started* manuals for SW300 Installation.

Most add-on options require only one person.

3.1.2 Tools Needed for Installation

The following tools could be needed during the installation of your controller subsystem. Not all of the tools listed are required for every cabinet type.

- Wrench to lower and tighten the four cabinet leveler feet.
- Allen wrench (5/32-inch) to open the front door of SWXSC-Dx-series cabinet.
- Small straight-edge screwdriver to install host port cables.
- ESD wrist strap for handling the controller or cache modules (if applicable). The part number for the Portable Anti-Static Kit is 29-26246-00.
- A pointed object for pushing the port buttons on the operator control panel.

3.1.3 Electrostatic Discharge Protection

This section describes the necessary precautions and procedure for protecting the controller subsystem components against electrostatic discharge (ESD). ESD is a common problem for any electronic device and may cause lost data, system down time, or other problems. The most common source of static electricity is the movement of people in contact with carpets and clothing materials. Low humidity enables a large amount of electrostatic charge to build up.

Use the following strategies to minimize electrostatic discharge problems:

- Maintain more than 40 percent humidity in the room where your subsystem resides
- Place the subsystem cabinet away from heavy traffic paths.
- Do not use carpet, if possible. If carpet is necessary, choose an antistatic carpet. If a carpet already is installed, place antistatic mats around the subsystem to help decrease electrostatic discharge.

CAUTION

Use proper ESD grounding procedures or damage may result to your controller or cache modules.

Specific safety precautions must be taken when handling write-back cache modules. Therefore, only qualified service personnel may install or replace write-back cache modules.

ESD Grounding Preparation

Prior to handling (removing or replacing) a controller module or cache module, do the following:

1. Obtain and attach an ESD wrist strap to your wrist.
2. Plug or clip the other end of the ESD wrist strap to an appropriate grounding point on the storage subsystem cabinet. Grounding studs are usually located on one of the cabinet's vertical chassis rails.
3. Obtain an approved antistatic bag and/or a grounded antistatic mat.

3.1.4 Controller Components Handling Guidelines

As with any electronic equipment, some components of you controller subsystem need special handling. The following sections describe handling guidelines for modules, program cards, and cables.

3.1.4.1 Module Handling Guidelines

When handling controller or cache modules, use the following ESD grounding guidelines:

CAUTION

Use the ESD grounding procedure when handling a controller or cache module, or damage to the modules could result.

Specific safety precautions must be taken when handling write-back cache modules. Therefore, only qualified service personnel may install or replace write-back cache modules.

- Obtain and attach an ESD wrist strap to your wrist. Make sure the strap fits snugly to your wrist.
- Plug (or clip) the other end to your cabinet's grounding stud (or other chassis grounding point).
- Remove the module from its controller shelf slot and place it into an approved antistatic bag or onto a grounded antistatic mat. Remain grounded while working with the module on the antistatic mat.
- Remain grounded while installing a replacement module.
- Remove the ESD connection from the cabinet ground stud or other chassis grounding point.
- Remove the ESD wrist strap from your wrist.

3.1.4.2 Program Card Handling Guidelines

Use the following guidelines when handling the program card:

CAUTION

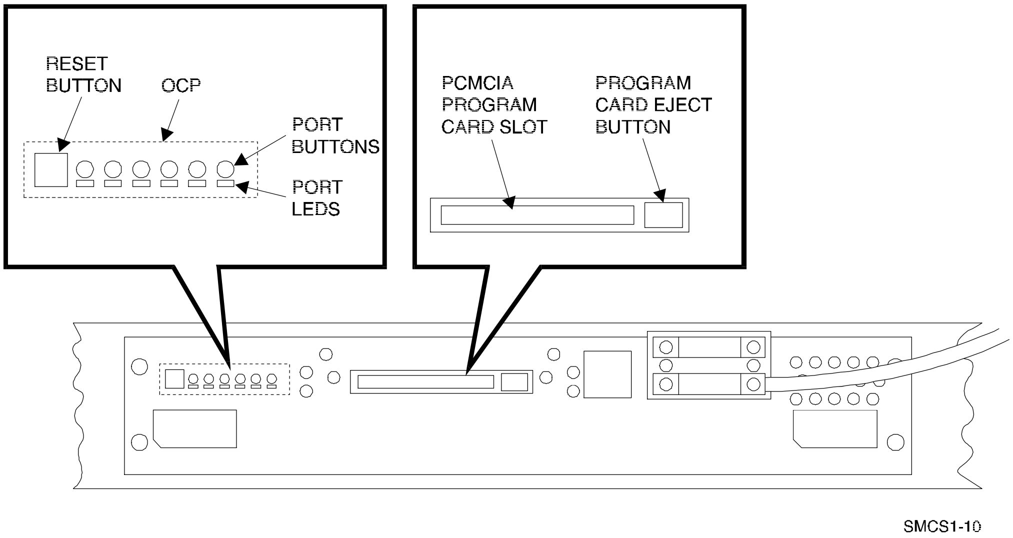
Follow these program card guidelines or damage to the program card may result.

The program card ESD shield must remain installed over the program card during controller operation to avoid electrostatic discharge that may cause the contents of the program card to be erased.

- Keep the program card in its original carrying case unless installing it.
- Do not twist or bend the program card.
- Do not touch the contacts.
- Keep out of direct sunlight.
- DO NOT immerse the program card in water or chemicals.

- Always push the eject button to remove the card. (See Figure 3-1.)
- An ESD strap is required for installation and removal of the card.

Figure 3–1 Location of Program Card Eject Button



3.1.4.3 SCSI Host Port Cable Handling Guidelines

SCSI host port cables can be removed or replaced with power applied. However, make sure that if you must remove the SCSI host port cables for any reason while power is applied, that you do the following:

- Halt activity on the host path to the target controller before servicing its host cables.
- If it is at the end of the SCSI host bus, leave the SCSI host port cable and the terminator connected to the trilink when you remove the trilink from the controller's front bezel.
- If it is in the middle of the SCSI host bus, leave both SCSI host port cables connected to the trilink when you remove the trilink from the controller's front bezel.

These actions are necessary to prevent breaking the SCSI bus connection. Be careful not to bend any connector pins when plugging the SCSI host port cables into the trilink.

NOTE

Hint: Use a very small straight blade screwdriver when disconnecting the trilink connector block (with cables and/or terminators attached) from the front bezel of the controller. The clearance between a terminator and a host port cable or two host port cables is minimal.

Refer to the *StorageWorks Solutions Configuration Guide* and the *StorageWorks Solutions Shelf and SBB User's Guide* for more information about SCSI bus cables and connectors, terminators, and trilink connector blocks.

3.1.5 Site Preparation

Site planning and preparation are necessary before installing an SWXRC-04 controller subsystem.

> **WARNING**
>
> To prevent damage to equipment and personnel, make sure all power sources meet the specifications required for this equipment. For more information, refer to the appropriate *Getting Started* manual(s) for SW300 Installation.

Your site preparation plan should include:

- Power requirements
- Floor space requirements
- Environmental considerations (such as temperature and humidity)
- Device environment (including maximum altitude for operation and storage)
- Subsystem weight considerations (for floor supports)
- Upgrade considerations (for future subsystem expansion)

Using a site preparation plan helps you fulfill the requirements to support your controller subsystem. After verifying that all requirements are met, you are ready to install your controller subsystem.

3.1.5.1 Power and Power Cord Requirements

Before installing your controller subsystem, ensure that the correct power cable for your site is attached to your StorageWorks cabinet, and that the power requirements for your country and your site have been met at the cabinet level.

For specific information about power cord plugs, refer to the StorageWorks cabinet specific installation and user's guides listed in this chapter.

3.1.5.2 Shelf Power Configuration Rules

Refer to the manual for your specific SBB shelf for specific power unit configuration rules. The term power unit describes both power supplies and battery backup units.

3.1.5.3 Environmental Considerations

The SWXRC-04 controller subsystem operates in a business or light industrial environment that complies with FCC Class A computing device standards. The cleanliness of the site is important for the operation of any computer system, and SWXRC-04 controllers require adherence to cleanliness standards. Temperature and humidity standards must be met to maintain proper operation of your subsystem. Refer to Section 1.5 for the environmental specifications for the SWXRC-04 controller.

3.2 Unpacking and Placing the Subsystem

When delivered, your controller subsystem is packed in a carton and attached to a shipping pallet. Upon receipt of your subsystem, perform the following tasks:

1. Check the carton and pallet for signs of shipping damage.
2. Report any damage to your vendor and to the local carrier that delivered your equipment.
3. Unpack and remove your subsystem cabinet from the shipping pallet. Refer to the cabinet-specific installation guides for unpacking instructions.
4. Keep all packing materials and shipping labels for later use and reference.
5. Remove and read your subsystem documentation before continuing with the installation process.
6. Move the subsystem cabinet into place at your site.
7. Open the front and back doors of the cabinet.
8. **If you are installing a preconfigured or CTO subsystem**, visually inspect all subsystem components to ensure that:
    - All cables are seated properly
    - All SBBs are seated properly
    - All controller and cache modules are seated properly
    - All shelf fans are seated properly
    - All program cards are loaded properly

    If you are installing a subsystem made up of individual components you have ordered, physically configure the interior of your cabinet to contain your power distribution units, power supplies, storage device shelves, and storage SBBs. See Chapter 4 for configuration rules for your cabinet.
9. Plug the cabinet power cord into the proper wall outlet for your subsystem's power requirements.

3.3 Connecting the Maintenance Terminal

To enable you to define your subsystem’s initial configuration parameters, connect a maintenance terminal to the EIA-423 terminal port on the front bezel of your controller as follows:

> **NOTE**
>
> You do not need a hardware maintenance terminal for normal operations. However, you must use a hardware maintenance terminal for initial controller parameter configuration.

1. Make sure the power switch on the back of the terminal is OFF (0).
2. Connect one end of the terminal cable to the back of the terminal.
3. Connect the other end of the terminal cable to the MMJ EIA-423 terminal port on the controller's front bezel.
4. Turn the terminal power switch to the ON position.
5. Set the terminal at 9600 baud, with 8 data bits, 1 stop bit, and no parity. Refer to your terminal documentation for terminal setup instructions.
6. Press the Return key if no prompt is visible on the screen. This brings you to the controller's command line interpreter (CLI) prompt.

> **NOTE**
>
> Your CLI prompt is factory-set to reflect your controller model, such as swxrc>. *Appendix B* provides details on how to change the prompt.
>
> This guide uses the designation "CLI>" to identify the CLI prompt.

3.4 Checking the Controller's Initial Parameters

1. Switch the circuit breakers (CB1) on your controller subsystem cabinet's cable distribution units to the ON (|) position. The subsystem controllers and devices begin their normal initialization sequence.

> **NOTE**
>
> Set the configuration parameters before connecting your host port cables.

2. From your maintenance terminal, check to see which controller configuration parameters have been preset at the factory. A factory configuration printout could accompany your documentation packet. You can verify the factory preset parameters by entering the following commands at the CLI> prompt:

```
CLI> SHOW THIS_CONTROLLER
```

or

```
CLI> SHOW OTHER_CONTROLLER
CLI> SHOW DEVICES
```

If initial parameters are not set, see Chapter 5 for the specific order for setting parameters for nonredundant and dual-redundant controller configurations.

NOTE

Whenever you restart the controller, device activity LEDs momentarily light while the controller discovers what devices are attached. Do not attempt to enter CLI commands that alter the device configuration (such as DELETE, SET, ADD, and so on) until after this discovery phase. This phase may take up to 2 minutes to complete.

After setting your initial parameters and defining your device configurations, print them out and keep the printout available to assist in servicing the subsystem in the future. Make a new printout each time you change your configuration parameters or add or delete units or storagesets.

IMPORTANT WRITE-BACK CACHE NOTE

If your system contains write-back cache modules, their batteries were completely charged at the factory. It is normal for the batteries to have discharged slightly in shipment. To ensure absolute data integrity, the advanced write-back cache and RAID features of your controller require fully-charged batteries to operate. These advanced features may not be available immediately after installation, until the batteries have had an opportunity to completely recharge.

3.5 Connecting the Host Port Cables

Connect host port cables, trilink connector blocks, and terminators to your controller as follows:

> **NOTE**
>
> Do not connect the controller end of any internal host port cables to your controller unless the controller's initial parameters have been set and the host port SCSI IDs have been set.

1. Plug the supplied trilink connector block (part number H885-AA) into the host port connector on the front bezel of your controller (if it is not already in place). Figure 3-2 shows the H885-AA trilink connector block.

   Figure 3-3 shows the connection of a SCSI host port cable and terminator to the trilink connector block on a nonredundant SWXRC-04 controller in an HA-000 shelf.

   Figure 3-4 shows the connections for a dual-redundant configuration. In this configuration, a trilink connector block must be installed on each controller. Connect the two trilink connector blocks together with a short jumper cable, as shown.
2. Plug the SCSI host cable onto either of the free connectors on the trilink connector blocks. For dual-redundant controllers, the host cable can be connected to either trilink connector block.

Figure 3–2 Trilink Connector Block

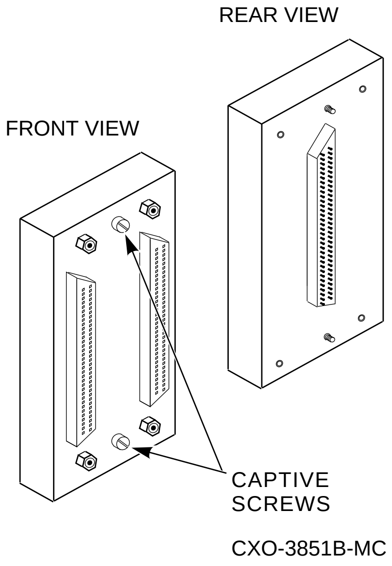
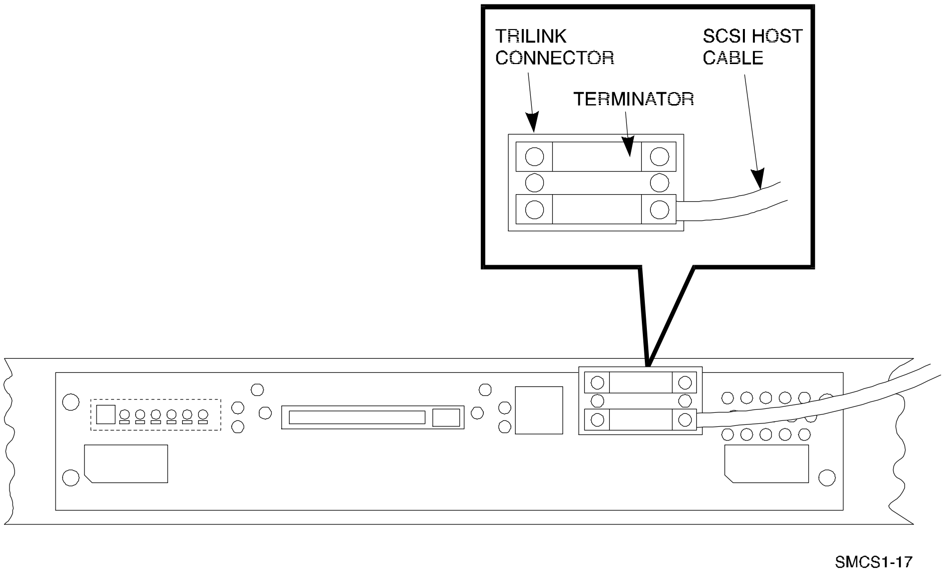


Figure 3–3 Host Port Cable Connection – Nonredundant Configuration



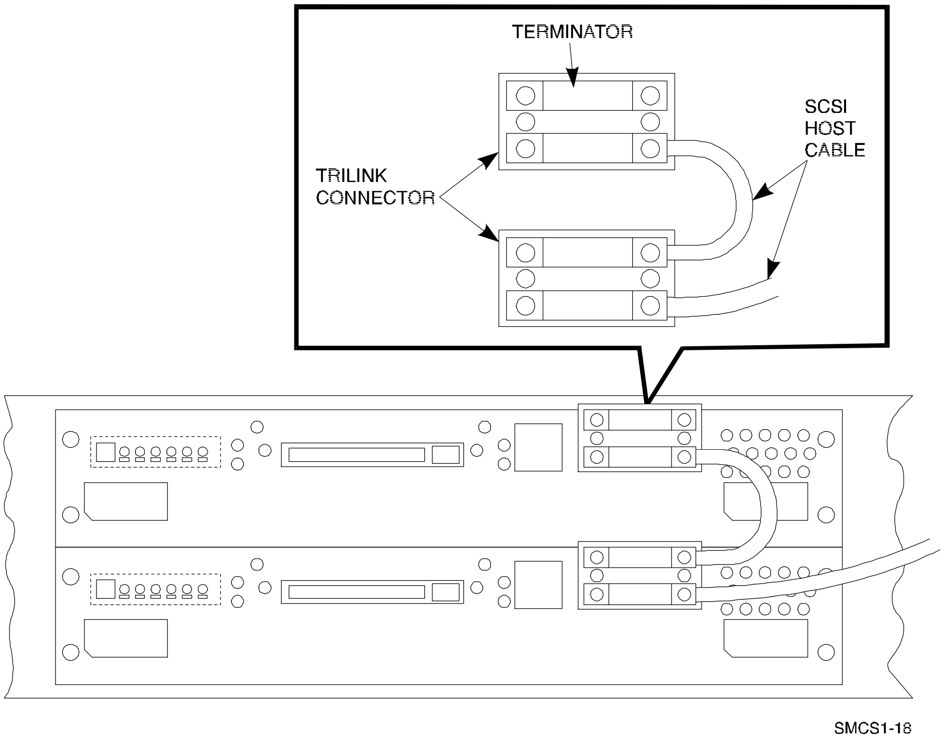
3. If the controller is to reside at the end of the host bus, the bus must be terminated at the controller. Plug the supplied terminator (option number H879-AA) onto the free connector on the trilink connector block. If the bus is to continue to other units, connect the appropriate SCSI bus cable instead.

   For dual-redundant controllers, plug the terminator or bus continuation cable onto the trilink connector block that is free.

4. Route the host port cable directly from the controller's trilink connector, through the cabinet, and to the host port adapter.

Two important considerations for all SCSI buses are bus termination and bus length. Each bus must be terminated at each end of the bus. Maximum bus lengths must be taken into consideration when designing your subsystem configuration.

Figure 3–4 Host Port Cable Connection – Dual-Redundant Configuration



4

Physical Storage Subsystem Configuration

This chapter describes physical aspects of the Storage Subsystem.

This chapter describes rules and restrictions as they apply to the physical configuration and connection of the following SWXRC-04 controller subsystem hardware:

- Cabinets
- Shelves
- Devices
- Controllers
- Hosts

This chapter describes physical configurations with respect to both standard and customized subsystems. Further information can be found in the specific StorageWorks cabinet and shelf documentation.

4.1 Physically Configuring StorageWorks Cabinets

The following sections present information to keep in mind when loading the controller and storage shelves in the cabinet. The design of the deskside RAID enclosure is such that the single storage enclosure is internal and fixed. Configuring shelves in the cabinet essentially is not required.

4.1.1 SWXSC-Dx-Series Data Center Cabinet

This section presents the rules to apply to subsystem configurations in SWXSC-Dx-series data center cabinets.

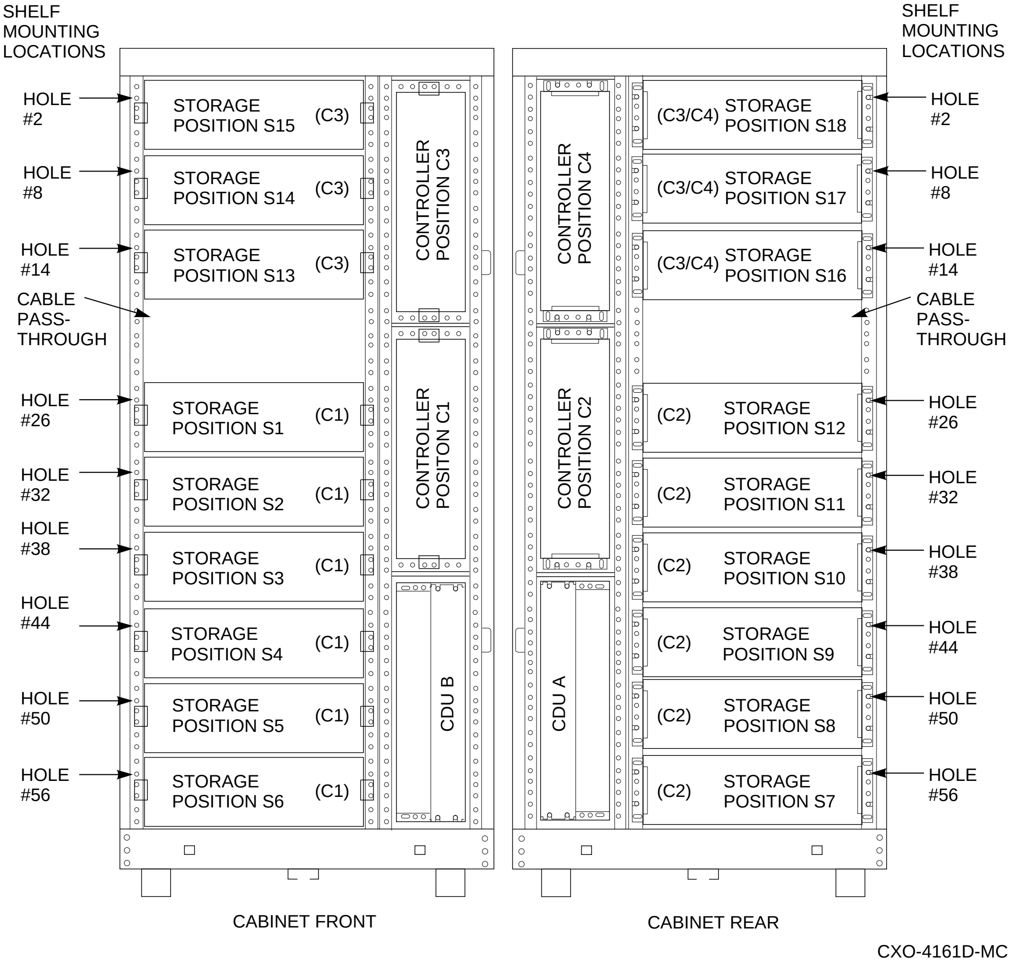
Refer to the *StorageWorks Solutions SW800 Data Center Cabinet Installation and User's Guide* for more details.

NOTE

In Figures 4-1 and 4-2, "S" indicates an SWXSS-02 storage shelf, and "C" indicates an SWXSS-01 controller shelf.

Figure 4-1 shows the loading sequence for storage and controller shelves in an SWXSC-Dx-series data center cabinet.

Figure 4–1 SWXSC-Dx Series Data Center Cabinet Loading



- Standard shelf configuration

  Digital suggests a standard of three (or four) SWXSS-01 shelves connected to 18 SWXSS-02 shelves in a single SWXSC-Dx-series data center cabinet.

- Two device shelves per port (jumpered pairs)

  Two SWXSS-02 shelves can be joined on the same controller port with the following restrictions:

  - The SCSI-2 cable to the first SWXSS-02 storage shelf is 1.0 meter or less.[1]
  - The SCSI-2 cable from the first SWXSS-02 shelf to the second shelf is 0.5 meters or less. This requires two shelves to be immediately adjacent to each other.
  - The first SWXSS-02 storage shelf is configured for an unterminated single SCSI cable.

- Use of an upper controller shelf

  By convention, controller shelf C3 would use (only) the top three (or four) storage shelves in the front of the cabinet; the fourth controller shelf (C4) would use the top three (or four) storage shelves in the back of the cabinet.

- Number of devices

  Up to 42 devices can be attached using 7 3½-inch SBBs in each of 6 SWXSS-02 shelves attached to controllers with 6 controller ports.[2]

- Maximum number of device shelves

  Up to 18 horizontal SWXSS-02 device shelves are allowed.

- Vertical device shelves

  Vertical shelves are not used for device shelves because some devices require horizontal alignment. If desired, vertical shelf locations may be used for most disk drives. Refer to the device-specific documentation for requirements. (Any of the vertical shelves may be used. However, Digital recommends surrendering controller positions C4, then C3, first for storage shelves. Refer to Figure 4-1.)

---

[1] The associated SWXSS-01 controller shelf must be located near enough to satisfy this restriction.

[2] Redundant power and dual-redundant controllers are not supported when using 42 devices. This is not a recommended configuration.

4.1.2 SWXSC-Cx-Series Departmental Cabinets

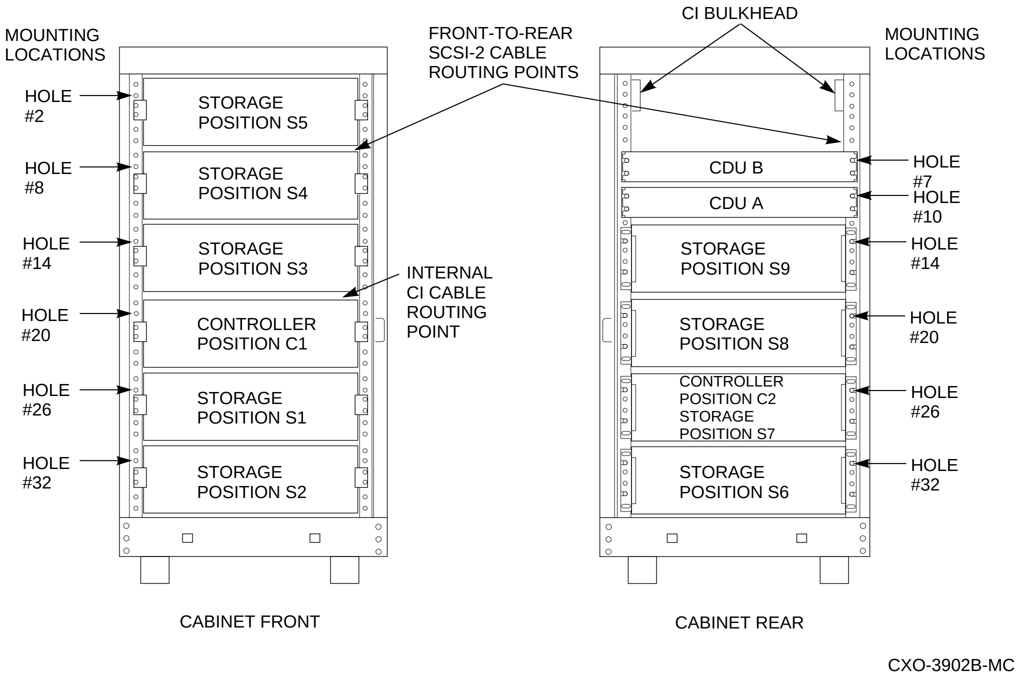
The rules presented in this section apply to subsystem configurations in SWXSC-Cx-series departmental cabinets. Refer to the *StorageWorks Solutions SW500 Cabinet Installation and User's Guide* for more details.

Figure 4-2 shows the loading sequence for storage and controller shelves in an SWXSC-Cx-series departmental cabinet.

- Standard shelf configuration

  A standard of one SWXSS-01 controller shelf connected to six SWXSS-02 storage shelves in a single SWXSC-Cx-series departmental cabinet is suggested.

Figure 4–2 SWXSC-Cx–Series Departmental Cabinet Loading



- Two SWXSS-01 shelves can be housed with a maximum of four SWXSS-02 shelves each as two subsystems.

- Two device shelves per port (jumpered pairs)

  Two SWXSS-02 shelves can be joined on the same controller port with the following restrictions:

  - The SCSI-2 cable to the first SWXSS-02 storage shelf is 1.0 meter or less.[3]
  - The SCSI-2 cable from the first SWXSS-02 shelf to the second shelf is 0.5 meters or less. This requires two shelves to be immediately adjacent to each other.
  - The first SWXSS-02 storage shelf is configured for unterminated single SCSI.
  - Controller shelf position C1 can be used with the pairs S1-S2 and S3-S4, and controller shelf position C2 can be used with the pair S8-S9, to satisfy these restrictions.

- Use of a second controller shelf

  By convention, the first controller shelf (C1) would use positions S1-S5; the second controller shelf (C2) would use positions S6, S8, and S9. This permits two subsystems, one with up to 24-28 3½-inch SBB devices (in the front), and the other with 18-21 3½-inch SBB devices (in the rear).

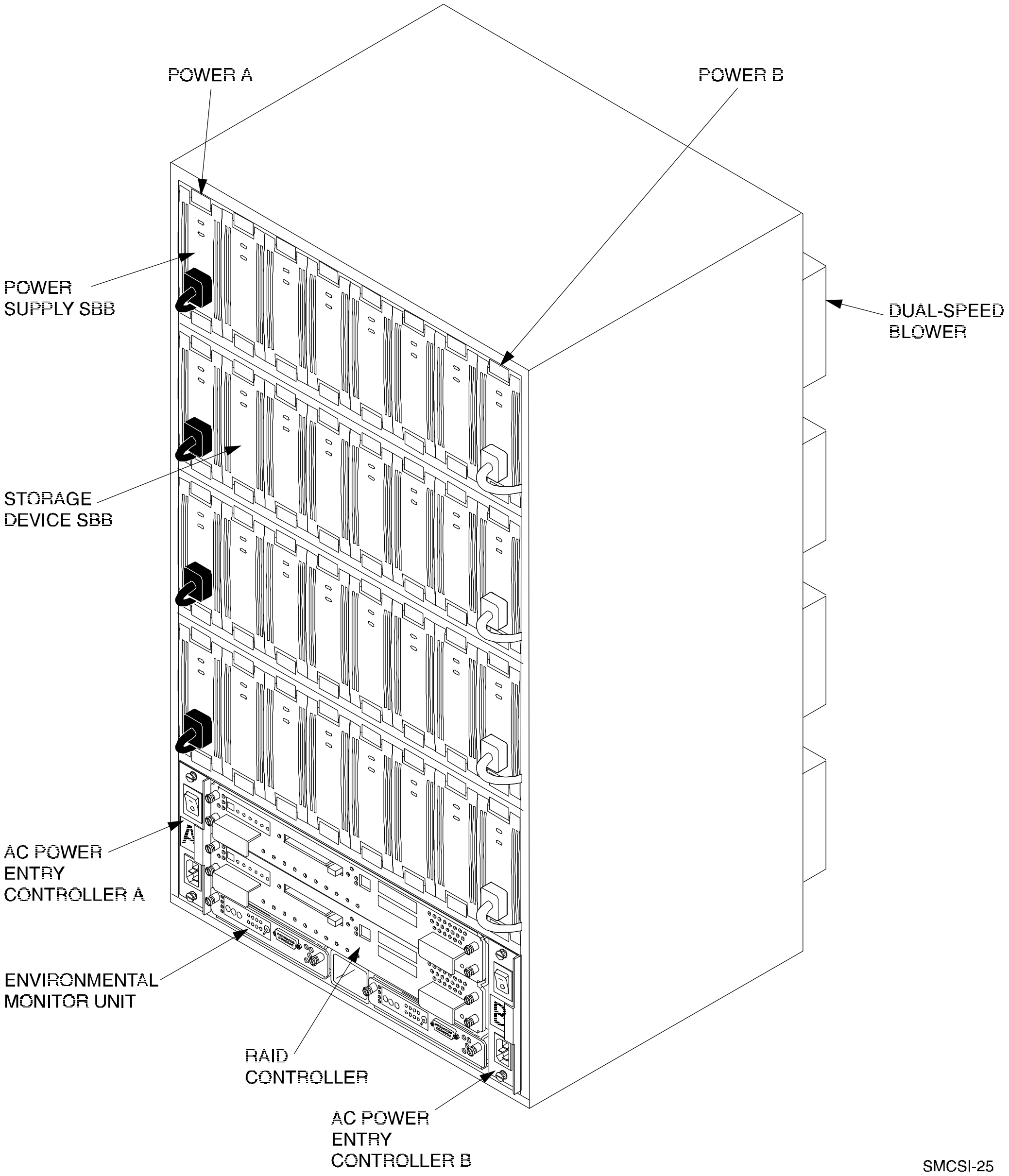
4.1.3 SWXSC-AA Office Expansion RAID Enclosure

The Office Expansion RAID enclosure, as shown in Figure 4-3, is a modular, deskside, StorageWorks enclosure designed to hold one integrated set of subsystem components. The enclosure comprises an SWXSC-AA storage enclosure mounted in the SWXSC-AA cabinet. The storage enclosure supports the equivalent of four SWXSS-02 SBB shelves and one SWXSS-01 controller shelf. The integrated design of the storage enclosure virtually eliminates the need for complicated physical configuring by the operator.

---

[3] The associated controller shelf must be located near enough to satisfy this restriction.

Figure 4–3 SWXSC–AA Office Expansion RAID Enclosure



The SWXSC-AA Office Expansion RAID enclosure accepts only the SWXSC-AA storage enclosure; it does not accept other shelf types. See the *SWXSC-AA Office Expansion RAID Enclosure User's Guide* for more information on the enclosure. In this guide, the SWXSC-AA Office Expansion RAID enclosure is called the "SWXSC-AA cabinet." and refers to the combination of the SWXSC-AA cabinet and storage enclosure.

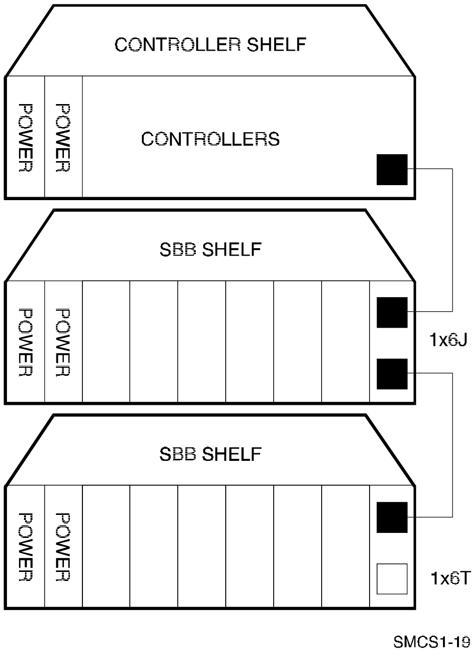
4.2 Physically Configuring StorageWorks SBB Shelves

The following configuration rules apply to the arrangement of controller and device shelves.

SWXSS-Series SBB Shelves – SWXSS-series shelves may be arranged in any legal SCSI-2 configuration, subject to the following:

- No more than a single extension joining two SWXSS-02 SBB shelves is permitted. The two SWXSS-02 shelves must be physically adjacent to each other. Figure 4-4 shows an example of device shelves in a single extension configuration.
- Connecting a 1.0 meter cable from a controller shelf to a device shelf allows for device shelf jumpering. Connecting a 2.0 meter cable does not permit shelf jumpering. (Required cable length varies depending on cabinet type, device shelf position, and controller shelf position.)

Figure 4–4 Single Extension from Device Shelf to Device Shelf



4.3 Device SBB Configuration in the SWXSS-02 Shelf

The following sections describe recommended device configurations for 3½-inch SBBs in SWXSS-02 shelves.

4.3.1 3½-Inch SBB Configuration Restrictions

The only restriction for adding 3½-Inch SBBs to SWXSS-02 SBB shelves is to prevent SCSI ID conflicts. Devices should not be installed in slot 6 (SCSI ID 6) of an SWXSS-02 shelf when a controller serving the shelf is in controller slot 6 (the slot closest to the SCSI bus cables) of the controller shelf.

Refer to your SPD and release notes for a list of specific supported device types.

4.3.2 Shelf Configuration Tables

Table 4-1 provides the information necessary to configure and install 3½-inch SBBs in SWXSS-02 SBB shelves.

4.3.2.1 Table Conventions

Following are the conventions used in Table 4-1. The designation shows the possible devices in each shelf and the possible number of devices in similarly configured shelves.

(n)mxoT

(n)mxoJ

where:

n is the number of device shelves.
m is the number of SCSI-2 connections to a device shelf.
o is the number of devices on each SCSI-2 connection.
T indicates the device shelf is terminated.
J indicates the device shelf is jumpered.

According to the formula:

*m*o* = possible devices in each shelf.

*n*m*o* = possible number of devices in similarly configured shelves.

4.3.2.2 3½-Inch SBB Configurations

Table 4-1 lists some recommended configurations for 3½-inch SBBs in SWXSS-02 shelves.

Table 4–1 3½-Inch SBB Configurations, 6-Port Controller

Number of Devices	Number of SWXSS-02 Shelves*	Configure as**	Available for 3½-Inch SSBs***	Ports Used
1-2	1	(1)2x3T	5-4	1-2
3-4	2	(2)2x3T	9-8	3-4
5-18	3	(3)2x3T	13-0	5-6
19-24	4	(2)2x3T (2)1x6T	5-0	6
25-30	5	(1)2x3T (4)1x6T	5-0	6
31-36	6	(6)1x6T	5-0	6
37-42****	6	(6)1x7T	5-0	6

Notes

2x3T: Two (split) SCSI-2 connections, separately terminated in the shelf. The devices appear as IDs 0, 2, 4, and 1, 3, 5.

1x6T: Single path SCSI-2 connection terminated in the shelf. The devices appear as IDs 0 through 5.

1x7T: Single path SCSI-2 connection terminated in the shelf. The devices appear as IDs 0 through 6.

* Consult the StorageWorks Solutions Shelf User's Guide for SWXSS-02 shelf information.

** Each SWXSS-02 shelf's upper SCSI-2 port connector is cabled to a controller port. The lower SCSI-2 port connector is attached to a controller port for 2x3T configurations and is unused for a 1x6T or 1x7T.

*** Available for future expansion.

**** Nonredundant controller and power (not recommended).

4.3.3 Atypical Configurations

By unbalancing the number of devices per controller port, configurations can be devised with a smaller shelf count. This results in lower performance and/or availability. The minimum SWXSS-02 shelf count for various numbers of 3½-inch SBBs is listed in Table 4-2 .

Table 4–2 Small Shelf Count Configurations, 6-Port Controller

Number of Devices	Number of SWXSS-02 Shelves*	Configure as	Ports Used
1-6	1	1x6T	1
7-12	2	1x6T	2
13-18	3	1x6T	3
19-24	4	1x6T	4
25-30	5	1x6T	5
31-36	6	1x6T	6
37-42**	6	1x7T	6

Notes

* Consult the StorageWorks Solutions Shelf User's Guide for SWXSS-02 shelf information.

** Nonredundant controller and power configurations (not recommended).

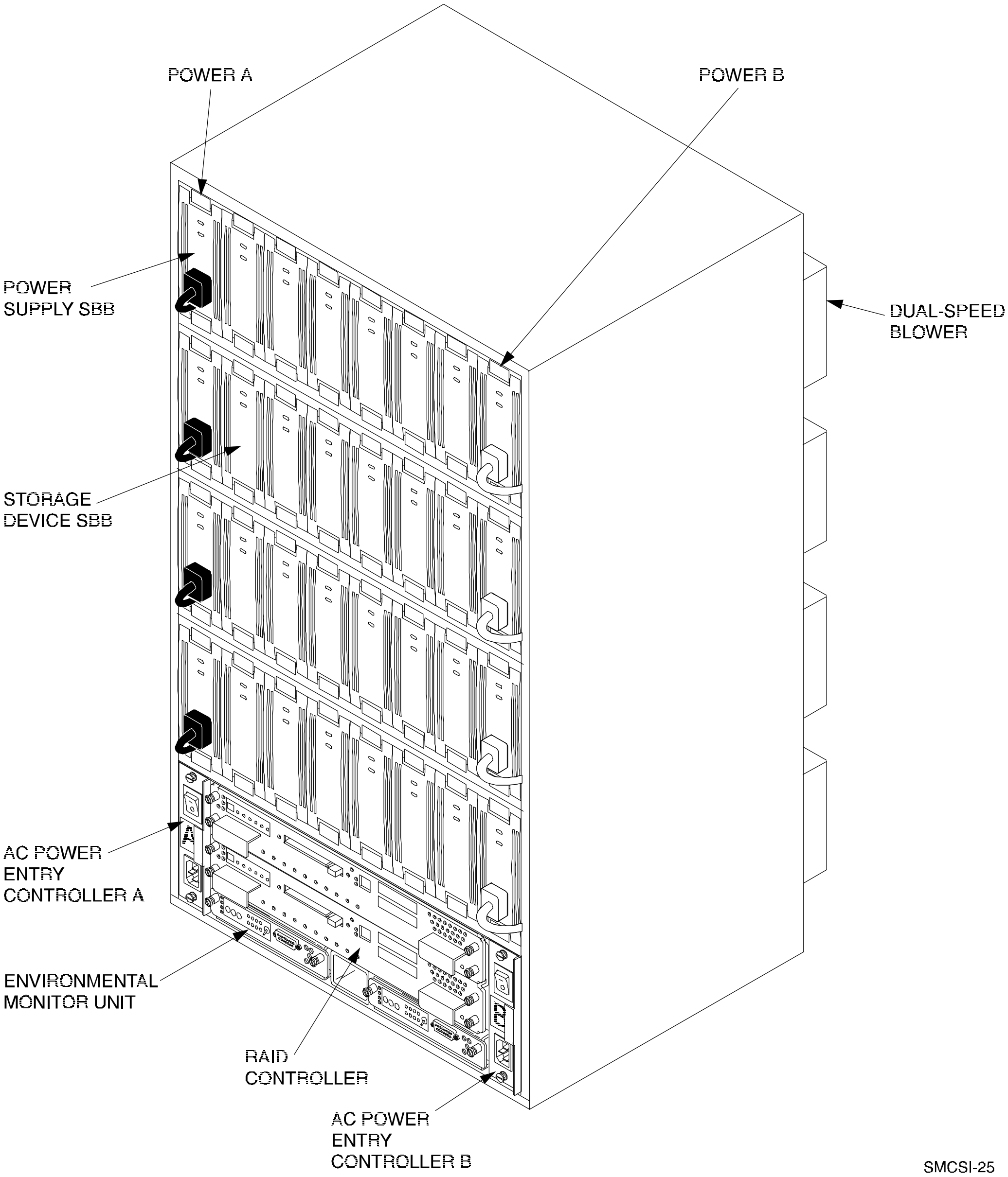
4.4 Device SBB Configuration in the SWXSC-AA Storage Enclosure

The following sections provide information on configuring and installing device SBBs in the SWXSC-AA storage enclosure.

4.4.1 SWXSC-AA Storage Enclosure Description

The SWXSC-AA storage enclosure is an integrated controller and storage shelf. The enclosure is standard width, but is approximately the height of five SWXSS-02 device shelves, as shown in Figure 4-5.

Figure 4–5 SWXSC – AA Storage Enclosure



The SWXSC-AA storage enclosure contains a single backplane for controller-to-storage connections. No external SCSI device cables are needed, and very little physical configuring is required. See the *StorageWorks SWXSC-AA Office Expansion RAID Enclosure User's Guide* for more information.

4.4.2 Configuring the Storage Enclosure

The SWXSC-AA storage enclosure (refer to Figure 4-5) can accommodate storage, controller, and support hardware within the limits described in Table 4-3.

Table 4–3 SWXSC-AA Storage Enclosure Configuration

Field Replaceable Unit	Minimum	Maximum
StorageWorks Building Block (SBB) Shelf Power Supplies	4	8
3½-Inch Storage SBBs	1	24
SWXRC-04 Controllers	1	2
Controller Cache Modules	1	2
Environmental Monitor Units (EMUs)	1	2
AC Power Entry Controllers	1	2
Dual Speed Blowers	8	8

The SWXSC-AA storage enclosure is called a "deskside RAID" configuration because its design allows for the easy creation of storagesets (stripesets, mirrorsets, and RAIDsets). Although you can configure 3½-inch SBBs in any combination within the SWXSC-AA storage enclosure, Digital recommends that you take advantage of the shelf layout for storagesets.

The SWXSC-AA storage enclosure includes an integrated backplane that contains six single-ended SCSI device buses. The backplane device buses eliminate the need for external device cabling to the controller device ports.

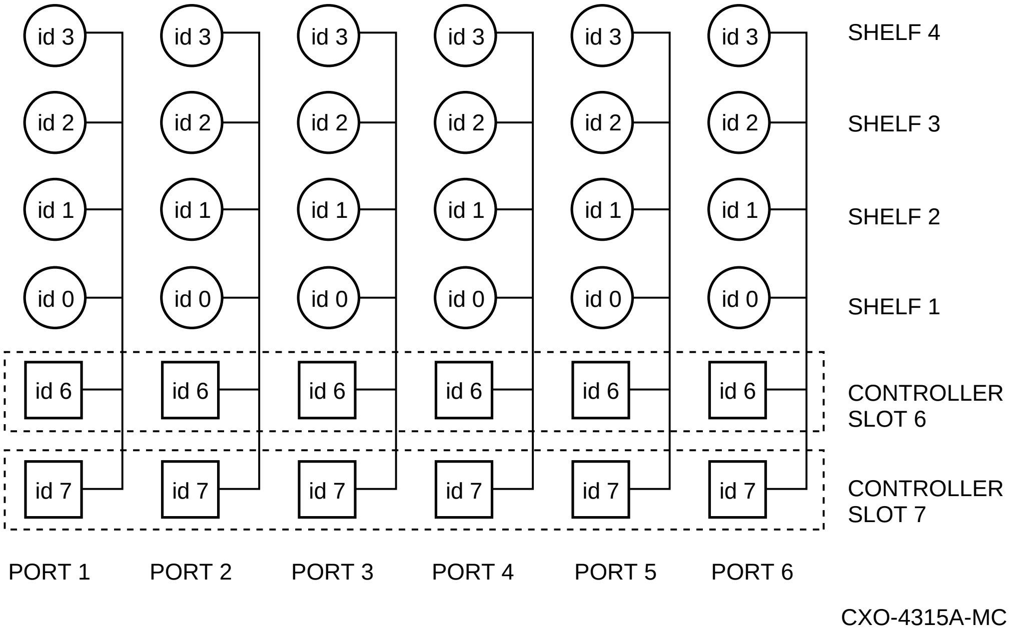
Each device bus runs vertically, attaching to up to four SCSI devices (IDs 0-3). The controller slots in the lower portion of the shelf occupy SCSI IDs 6 and 7, in much the same way as in a SWXSS-01 shelf. The SWXSC-AA storage enclosure SCSI device bus configuration appears in Figure 4-6.

NOTE

The vertical arrangement of the SWXSC-AA storage enclosure device buses marks a departure from the convention established with SWXSS-02 device shelves. Devices arranged horizontally in a SWXSS-02 shelf are typically on the same controller port. However, horizontally arranged devices in an SWXSC-AA storage enclosure are each on a *different* controller port.

This layout enables storagesets, which normally consist of devices on different controller ports (for best performance and availability), to be arranged as horizontal groups of devices.

Figure 4–6 SWXSC–AA Storage Enclosure SCSI Buses



4.5 Physically Configuring Controllers

This section describes the specifics of configuring StorageWorks controllers.

4.5.1 Nonredundant Controllers

The following guidelines apply to nonredundant controllers:

- A single controller must be installed in the slot furthest from a SWXSS-01 shelf's SCSI connectors. This slot is SCSI ID 7. By using SCSI ID 7, SCSI ID 6 (the other controller slot) is available as an additional ID on the SWXSS-02 device shelf.
- The maximum recommended controller subsystem configuration using SWXSS-series shelves is six devices per controller port. This allows for the addition of another controller, and additional power supplies in the storage shelves. A nonredundant controller configuration *can* support seven devices per port. However, Digital still recommends six devices per port to permit the ease of future upgrade.
- The maximum controller subsystem configuration using an SWXSC-AA storage enclosure is four devices per controller port. The maximum is determined by the design of the SWXSC-AA storage enclosure and backplane.
- The controller for a configuration using a SWXSC-AA storage enclosure must be installed in the lower portion of the shelf (refer to Figure 4-5). Controllers cannot be installed in the upper four areas, which are reserved for devices.

4.5.2 Dual-Redundant Controllers

The following guidelines apply to dual-redundant controllers:

- Dual-redundant controllers are located in the same shelf, and are connected to each other through the shelf backplane. Both controllers have access to all the devices on each other's ports. This setup increases availability and provides for failover when one controller in the pair fails. (The surviving controller takes over service to all devices.)
- Dual-redundant configurations follow the same guidelines as nonredundant configurations, except there is (for SWXSS-series shelves) no option to increase to seven devices per port.
- Both controllers' cache modules must be the same type (read/write-back), have the same number of megabytes, and both firmware versions must be identical.
- Dual-redundant SWXRC-04 controllers must be on the same host SCSI bus.

4.5.3 Optimal Performance Configuration

For optimal performance, configure to the following guidelines:

- Balance the number of devices on each port of a controller. For example, for 18 3½-inch SBBs, place 3 devices on each of 6 ports. This permits parallel activity on the controller's available ports to the attached devices. Figure 4-7 is an example of how to balance devices across ports.
- Evenly distribute higher performance devices across separate ports so that higher and lower performance devices are intermixed on the same port. This intermixing of higher and lower performance devices on the same port benefits overall performance. Use the guidelines in Table 4-4.
- Limit the number of devices per controller port to three in dual-redundant configurations. In doing so, both controllers access three devices per *each other's* port, maintaining six SCSI-2 devices combined total.

Figure 4–7 Balanced Devices Within Device Shelves

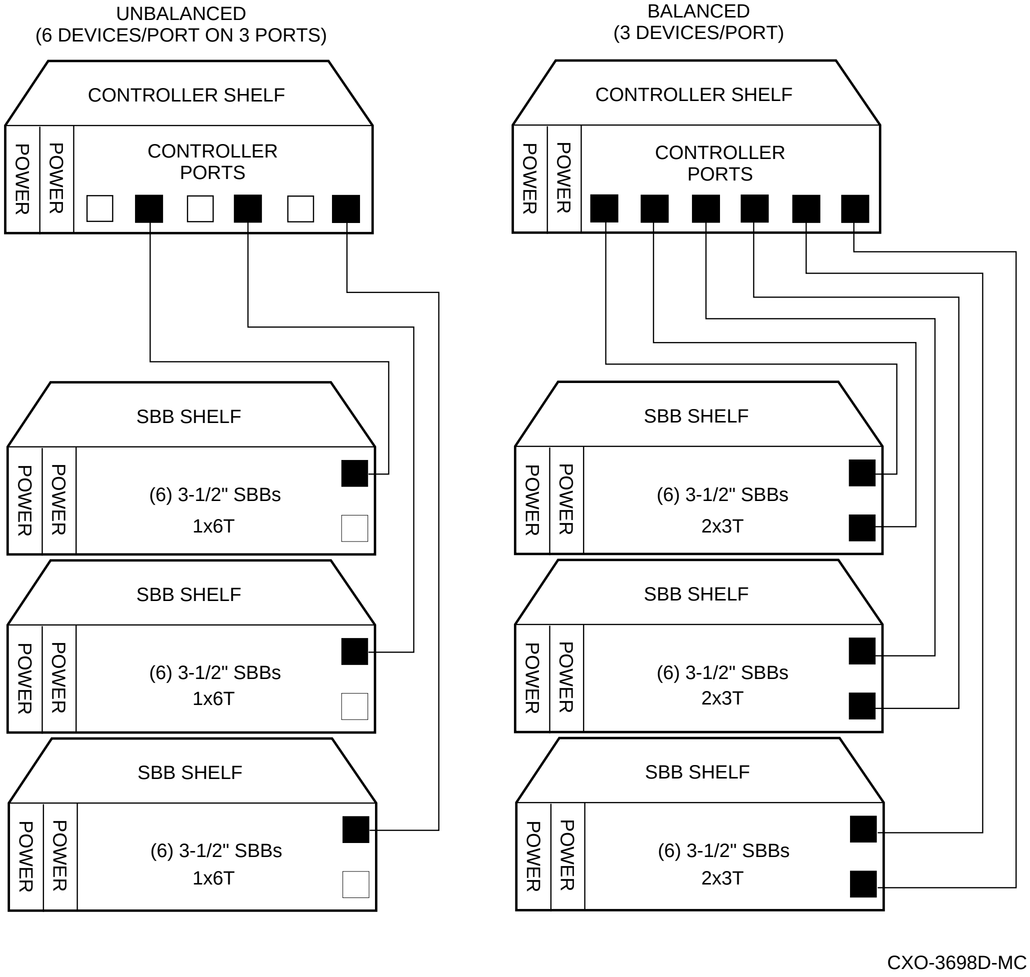


Table 4–4 High-Performance Devices per Port

Total High-Performance Devices	High-Performance Devices per Port (3-Port Controller)	High-Performance Devices per Port (6-Port controller)
1-3	1	1
4-6	1	2
7-9	2	3
10-12	2	4
13-15	3	5
16-18	3	6

Highest Performance

To obtain the highest performance possible, use a dual-redundant configuration and balance the number of devices across the two controllers. Do this through your operating system by ordering how devices are mounted or sequenced and by setting preferred path definitions.

Following this guideline results in reducing each controller's load to approximately half of the devices normally accessed through each controller. Should one controller fail, the surviving controller automatically assumes service to the failed controller's devices.

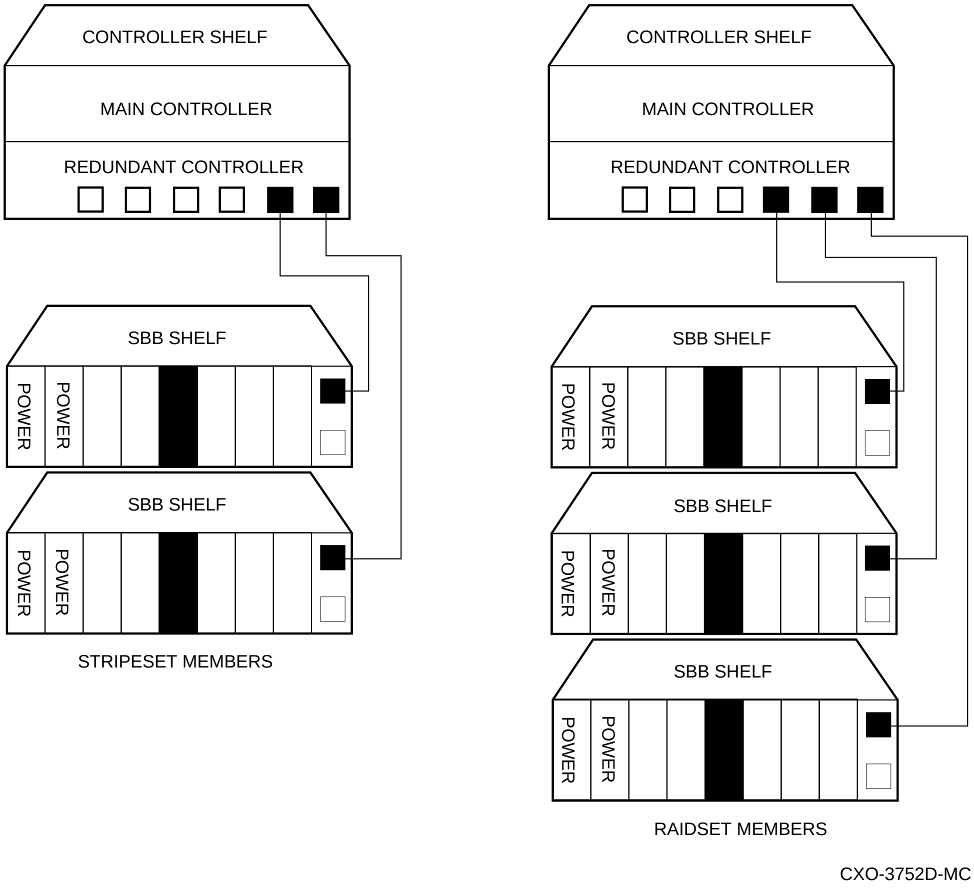
4.5.4 Optimal Availability Configuration

For optimal availability, configure to the following guidelines:

- Use dual-redundant controllers and redundant power supplies in all shelves.
- Place storageset members on different controller ports and different device shelves.
- Use predesignated spares on separate controller ports and device shelves.

Figure 4-8 shows examples of optimal configurations for RAIDset members and designated spares on separate controller ports.

Figure 4–8 Optimal Availability Configurations



Highest Availability

For highest availability, especially with RAID implementations, follow these guidelines:

- For host-based RAID implementations, split the normal access path between controllers.
- Use redundant power supplies in all shelves.

4.6 Host Considerations

The following section explains important cable considerations when configuring the SWXRC-04 controller and subsystem to the host CPU.

4.6.1 Host Cables

When configuring cables to and from the controller, do not exceed these maximum lengths (end-to-end):

- For 8-bit, single-ended bus cables, with a transfer rate of 5 MB/s, the maximum length is 6 meters (19.7 feet).
- For 8-bit, single-ended bus cables, with a transfer rate of 10 MB/s, the maximum length is 3 meters (9.8 feet).
- For 16-bit, differential bus cables, with a transfer rate of 20 MB/s, the maximum length is 25 meters (82.0 feet).

5

Operating the Storage System

This chapter presents the information necessary to logically configure and operate the SWXRC-04 controller subsystem. It covers normal and failover operation, and operating system support.

> **NOTE**
>
> The “configurations” discussed in this chapter are those set by the operator, employing the CLI. Refer to Chapter 4 for *physical* configuration of the subsystem hardware.

5.1 General Sequence of Configuration and Operation

The general sequence for using the SWXRC-04 controller is as follows:

1. Establish a maintenance terminal connection with the controller and familiarize yourself with the command line interpreter and operator control panel.
2. Initialize the controller and subsystem.
3. Set the controller's initial operating parameters.
4. Set the logical configuration of the subsystem's storage devices.
5. Run an acceptance test to validate the integrity of the subsystem.

5.2 Maintenance Terminal

A maintenance terminal is used to communicate with the controller and the storage subsystem. The maintenance terminal may be a hardware device physically connected to the controller or a virtual terminal provided by the host's software.

You do not need a maintenance terminal for normal operation. However, you must connect a maintenance terminal for initial controller configuration. Thereafter, use either a maintenance terminal or a host (virtual) terminal to communicate with the controller.

5.2.1 Hardware Maintenance Terminal

A hardware maintenance terminal is a locally connected EIA-423 compatible terminal.

Follow this procedure to connect a maintenance terminal:

1. Make sure the power switch on the back of the terminal is in the off (0) position.
2. Connect one end of the terminal cable to the back of the terminal or PC.
3. Connect the other end of the terminal cable to the maintenance port on the controller.
4. Set your terminal at 9600 baud, 8 data bits, 1 stop bit, and no parity. Refer to your terminal documentation for terminal setup instructions.

5.3 Command Line Interpreter

The CLI is the firmware user interface to the controller. The CLI enables you to control storage and controller configurations through commands entered into the maintenance terminal. The following sections explain how to use the CLI, and how it defines and modifies configurations. A detailed description of CLI commands is provided in Appendix B.

5.3.1 Accessing the CLI

You can access the CLI through a hardware maintenance terminal (see Section 5.2).

To access the CLI through a maintenance terminal, connect the terminal and press the Return key. You must use a maintenance terminal to set the controller's SCSI ID and initial configuration.

Thereafter, you may use a virtual (host) terminal to modify the configuration. The method of establishing the virtual terminal connection varies depending on your operating system and interface.

NOTE

Your CLI prompt could be factory-set to reflect your controller model, such as `swxrc>` . Appendix B provides details on how to change the prompt.

This guide uses the designation "CLI>" to identify the CLI prompt.

5.3.2 Exiting the CLI

If you are using a maintenance terminal, you cannot exit the CLI. Entering the EXIT command merely restarts the CLI and redisplays the controller type and any last fail error information.

5.3.3 Command Sets

The CLI consists of the following command sets:

- Failover commands

  Failover commands support dual-redundant controller configurations.

- Controller commands
  - Set and show the basic controller parameters.
  - Set the controller ID (host SCSI target ID).
  - Set the resident terminal characteristics.
  - Restart the controller.
  - Run resident diagnostics and utilities (see Chapter 8).

- Device commands

  Device commands specify and show the location of physical SCSI-2 devices attached to the controller. Locations of devices are specified using their SCSI PTL designation.

  Only devices that have been defined by the ADD command are seen or used by the controller. Devices that have been placed in a shelf, but have not been added, are not automatically used by the controller. Use either the CONFIG or CFMENU utility to quickly add such devices (see Chapter 8).

- Storageset commands

  Storageset commands add, modify, rename, and show storagesets (stripesets, mirrorsets, and RAIDsets). These commands also apply (to some extent) to the spareset and failedset.

- Logical unit commands

  Logical unit commands add, modify, and show logical units built from devices and storagesets.

- Utility commands

  You can use the RUN command to invoke various exercisers and utilities such as DILX and CLONE. These are fully described in Chapter 8.

NOTE

Remember these two guidelines when using the CLI:

Not all configuration parameters need to be specified on one line. They can be entered by using multiple SET commands.

Only enough of each command need be entered to make the command unique (usually three characters). For example, SHO is equivalent to SHOW.

5.4 Operator Control Panel

You can use the operator control panel (OCP) to reset the controller, quiesce the SCSI-2 device buses attached to the controller, and interpret error conditions that result in LED error codes. The OCP and its use are described in *Chapter 7*.

5.5 Initialization

The following sections discuss the operating conditions related to initialization of the controller and subsystem.

5.5.1 Controller Initialization

The controller initializes after any of the following conditions:

- Power is turned on.
- The firmware resets the controller.
- The operator presses the green reset (//) button.
- Controller receives RESTART command.

NOTE

Keep the program card in its slot during controller subsystem operation. If you remove the program card, the controller resets.

See Chapter 8 for a description of the initialization of both the controller and its cache module. (The process is described in Chapter 8 because some of the initialization diagnostics are available as a controller self-test function for the operator.)

IMPORTANT WRITE-BACK CACHE NOTE

If your system contains write-back cache modules, their batteries were completely charged at the factory. It is normal for the batteries to have discharged slightly in shipment. To ensure absolute data integrity, the advanced write-back cache and RAID features of your controller require fully-charged batteries to operate. These advanced features may not be available immediately after installation, until the batteries have had an opportunity to completely recharge.

5.5.2 Dual-Redundant Controller Initialization

The controllers in a dual-redundant configuration run the same initialization sequence that is described in Chapter 8, except that they exchange signals during their individual initialization sequences. The first signal occurs after one controller starts initializing. The signal informs the other controller that an initialization is occurring. This way, the other controller does not assume that the initializing controller has malfunctioned and does not attempt to disable it.

5.5.3 Subsystem Initialization

Full StorageWorks subsystem initialization occurs when the subsystem is switched on for the first time. In the event of a reset due to one of the following conditions, a subset of the initialization sequence is run:

- A partial or complete power failure
- Equipment failure
- An error condition

A complete StorageWorks subsystem initialization includes the following:

1. When the subsystem is turned on, all shelves in the subsystem are reset. Then, entities in the shelves (including storage devices, controllers, and cache modules) run their initialization and self-test sequences.
2. During initialization, the controller interrogates the entities with which it has connections, including other controllers in the subsystem.
3. When the initialization sequence on all entities is completed, the controller begins data transfer and other operations with the host.

5.5.4 Initial Configuration (Nonredundant Controller)

After installation of a nonredundant controller, use the CLI to define its parameters in the following order from a maintenance terminal:

1. Enter the following command to set a valid controller ID:

   ```
   CLI> SET THIS_CONTROLLER ID=n
   ```

   Where *n* is the superset of unique SCSI target ID(s) (range 0-7, up to four total, as: SET THIS_CONTROLLER ID=w,x,y,z).

   > **NOTE**
   >
   > Always restart the controller after setting the ID.

2. Restart the controller either by pressing the green reset (//) button, or by entering the following command:

   ```
   CLI> RESTART THIS_CONTROLLER
   ```

3. Enter the following command to verify the preceding parameters were set:

   ```
   CLI> SHOW THIS_CONTROLLER FULL
   ```

4. Connect the host port cable to the front of the controller (see Chapter 9).

5.5.5 Initial Configuration (Dual-Redundant Controllers)

In a dual-redundant configuration, one terminal can set both controller configurations. After installation of both controllers, use the CLI to define the controllers' parameters in the following order from a maintenance terminal connected to one controller:

1. Enter the following command to set a valid controller ID:

   ```
   CLI> SET THIS_CONTROLLER ID=n
   ```

   Where *n* is the superset of unique SCSI target IDs (range 0-7, up to four total–SET THIS_CONTROLLER ID=*w*,*x*,*y*,*z*) assigned to *both* controllers. Then enter the following command:

   ```
   CLI> SET THIS_CONTROLLER PREFERRED_ID=n
   ```

Where *n* represents the unique SCSI target IDs from the superset (up to four total–SET THIS_CONTROLLER PREFERRED_ID=w,x,y,z) assigned to only *this* controller. The remaining Ids from the superset are automatically assigned to the companion controller.

CAUTION

The SET FAILOVER command establishes controller-to-controller communication and copies configuration information. Always enter this command on one controller only. COPY=*configuration-source* specifies where the good configuration data are located. *Never* blindly specify SET FAILOVER. Know where your good configuration information resides before entering the command.

2. Enter the following command to copy parameters to the other controller (the one *not* connected to):

   ```
   CLI> SET FAILOVER COPY=THIS_CONTROLLER
   ```

3. Restart both controllers either by pressing the green reset (//) buttons, or by entering the following commands:

   ```
   CLI> RESTART OTHER_CONTROLLER
   CLI> RESTART THIS_CONTROLLER
   ```

4. Enter the following commands to verify the preceding parameters were set.

   ```
   CLI> SHOW THIS_CONTROLLER FULL
   CLI> SHOW OTHER_CONTROLLER FULL
   ```

5. Connect the host port cables to the front of the controllers (see Chapter 9). Do not connect the two controllers in a dual-redundant pair to separate or different SCSI buses.

5.5.6 Logically Configuring Storage Devices

To automatically configure devices on the controller, use either the CONFIG or CFMENU utility described in Chapter 8.

To manually configure devices on the controller, use the following steps to add devices, storagesets, and logical units. The steps utilize the CLI and must be performed so that the host recognizes the added storage device

These steps may be run from a virtual terminal.

IMPORTANT WRITE-BACK CACHE NOTE

If your system contains write-back cache modules, their batteries were completely charged at the factory. It is normal for the batteries to have discharged slightly in shipment. To ensure absolute data integrity, the advanced write-back cache and RAID features of your controller require fully-charged batteries to operate. These advanced features may not be available immediately after installation, until the batteries have had time to completely recharge.

1. Add the physical devices by using the following command:

   ```
   CLI> ADD device-type device-name scsi-location
   ```

   For example:

   ```
   CLI> ADD DISK DISK100 1 0 0
   ```

   Where:

   - *device-type* is the type of device to be added. This is DISK.
   - *device-name* is the name to refer to that device. The name is referenced when creating units or storagesets.
   - *SCSI-location* is the PTL for the device. When entering the PTL, separate the three numbers with spaces.

2. Add the storagesets for the devices. Storagesets include stripesets, mirrorsets, and RAIDsets. See Appendix B for examples of adding storagesets. (If you do not want storagesets in your configuration, skip this step.)

   ```
   CLI> ADD RAIDSET RAIDS0 DISK100 DISK200 DISK300
   ```

3. Enter the following command to initialize the containers (devices, storagesets, or both) prior to adding logical units to the configuration.

   CAUTION

   The INITIALIZE command destroys all data on a container. See Appendix B for specific information on this command.

   ```
   CLI> INITIALIZE container-name
   ```

   Where *container-name* is a device or storageset that becomes part of a unit.

   When initializing a single-device container:

   - If NOTRANSPORTABLE (the default) was specified when the device was added, a small amount of disk space was made inaccessible to the host and used for metadata. The metadata is initialized.
   - If TRANSPORTABLE was specified, any metadata on the device is destroyed. See Appendix B for details on metadata and when INITIALIZE is required.

4. Add the units that use either the devices or the storagesets built from the devices by entering the following command:

   ```
   CLI> ADD UNIT logical-unit-number container-name
   ```

   Where:

   - *logical-unit-number* is the unit number the host uses to access the device.
   - *container-name* identifies the device or the storageset.

5.6 Acceptance Testing

After you install, set parameters for, and configure your controller, follow the guidelines in this section to acceptance test your subsystem.

1. Turn your system on. This resets all shelves and starts the spin-up cycle on devices within the shelves. This includes the initialization (diagnostics) on the controller(s) and device self-tests.
2. Run DILX using the default answers to the test questions (see Chapter 8). This tests all disk devices in your subsystem.

5.7 Failover

The following sections describe what failover is and how it works. For more information about the SET FAILOVER and SET NOFAILOVER commands, see Appendix B.

5.7.1 What is Failover?

Failover is the firmware process that takes place when one controller fails in a dual-redundant controller configuration. In a failover (dual-redundant) configuration, information is shared between the two controllers, such as:

- Storageset names (for example, mirrorsets, RAIDsets, and stripesets)
- Actual device configuration (PTL descriptions) and the association to named storagesets
- Logical unit definitions

NOTE

Both RAID array controllers in a dual-redundant pair “know about” each other's SCSI target IDs, but do not *share* them.

Prior to failover, resources are always accessible only to a particular controller because the controller is a SCSI target of the host, and LUNs must be visible through a single target ID. This is true whether or not a unit is currently interacting with the host.

In a failover configuration, all commands are shared between the two controllers with the exception of the following commands:

```
SET THIS_CONTROLLER
SET OTHER_CONTROLLER
SHOW THIS_CONTROLLER
SHOW OTHER_CONTROLLER
RESTART THIS_CONTROLLER
RESTART OTHER_CONTROLLER
SHUTDOWN THIS_CONTROLLER
SHUTDOWN OTHER_CONTROLLER
CLEAR INVALID_CACHE THIS_CONTROLLER
CLEAR INVALID_CACHE OTHER_CONTROLLER
```

In these cases, the command is directed to the correct controller:

- **THIS_CONTROLLER** refers to the controller to which the terminal is connected or which is the target of the virtual terminal connection.
- **OTHER_CONTROLLER** refers to the other controller in the dual-redundant pair.

NOTE

Devices, stripesets, RAIDsets, and mirrorsets are defined as **containers**. All changes in container configuration are automatically communicated between the two controllers.

5.7.2 Transparent Failover

Two SWXRC-04 controller modules installed in a StorageWorks controller shelf and connected to the same host SCSI bus operate as a redundant pair with transparent failover. Transparent failover operation is a process in which the transfer of storage subsystem control occurs in a manner transparent to the host.

Figure 5-1 shows two controllers connected in this manner. Each controller may be configured to have multiple SCSI target IDs, in one of five different configurations:

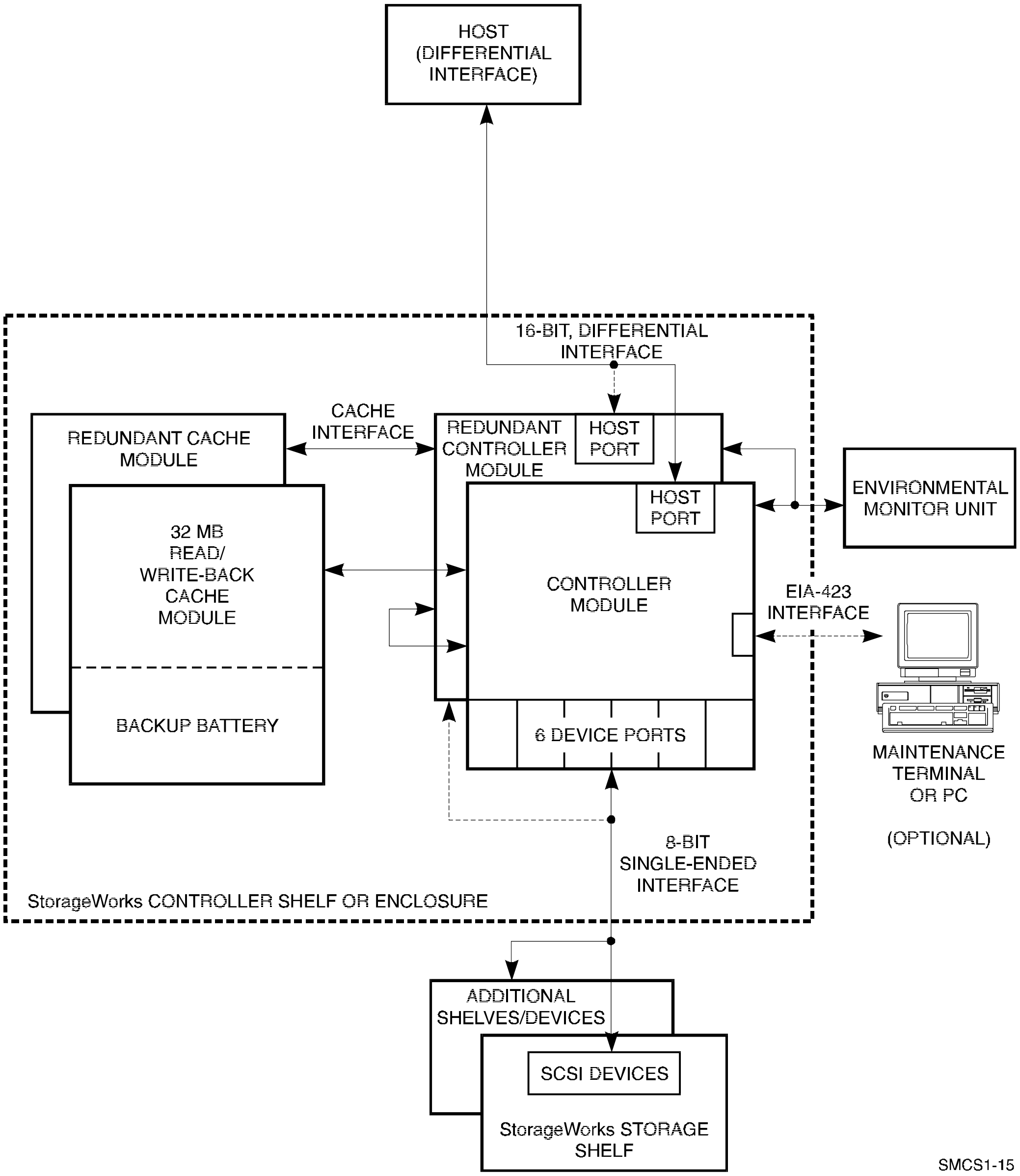
- Two controllers configured as a single, active, controller with a hot standby
- Each controller configured as an active controller, one servicing one target, the other servicing a different target
- Each controller configured as an active controller, one servicing one target, the other servicing two other targets
- Each controller configured as an active controller, one servicing one target, the other servicing three other targets
- Each controller configured as an active controller, one servicing two targets, the other servicing two other targets

In normal operation, each controller services only the targets it has been assigned. In a failover situation, the surviving controller services all of the targets from both controllers. Each target in any of these configurations supports up to eight LUNs.

CAUTION

Be aware that subsystem performance is better if you balance target IDs across your dual-redundant pair. Furthermore, if the hot spare controller in a 4–0 combination fails, and then the active controller fails, the hot spare does not assume service to any devices because it has already "secretly" failed.

Figure 5–1 Dual-Redundant Controller Configuration



5.7.3 Setting Failover

To place both controllers into a failover configuration, enter the following command:

```
CLI> SET FAILOVER COPY=configuration-source
```

Where *configuration-source* is either THIS_CONTROLLER or OTHER_CONTROLLER, depending on where the “good” copy of configuration information is located.

CAUTION

DIGITAL recommends that the controllers be set for failover *before* any device configuration commands are entered (for example, ADD DISK, ADD UNIT, or ADD RAIDset). Then, as devices, storagesets, and units are added to one controller's configuration, they are automatically added to the other controller's configuration.

IMPORTANT: Given two *unconfigured* controllers, it is possible to fully configure one controller, then enter the SET FAILOVER command. However, if the “wrong” controller is specified after the COPY=qualifier, all device configuration information will be *lost*. *Never* blindly specify SET FAILOVER. Always know where your “good” configuration information is located.

NOTE

Due to the amount of information that must be passed between the two controllers, the SET FAILOVER command may take up to 1 minute to complete.

When setting dual-redundant controllers for failover, make sure the target controller (the controller you are copying the configuration to) has no cache errors or unwritten cache data. Delete any remaining, configured units on the target controller to verify there are no cache errors before entering the SET FAILOVER command. Deleting units from the target controller will not create problems because, after setting failover, you will not be accessing those units anyway. (The target controller will only access the copied units.)

Observe the following considerations when setting dual-redundant SWXRC-04 RAID array controllers for failover:

- Subsystem performance will be better if you balance the assignment of target IDs across your dual-redundant pair.
- The controller you are copying configuration information to will automatically restart after entering the SET FAILOVER command.
- You MUST connect both controllers to the same host SCSI-2 bus. If you connect the controllers in a dual-redundant pair to different host buses, and one controller fails, the subsequent failover process will cause adverse effects on your system.

5.7.4 Using Failover Commands When Write-Back Cache Is in Use

Failover commands can be used while write-back caching is enabled provided the following restrictions are observed:

- When setting dual-redundant controllers for failover, make sure the target controller (the controller you are copying configuration data to) has no cache errors or unwritten cached data. Delete any units on the target controller to verify there are no cache errors, *before* entering the SET FAILOVER command.
- Do not take a dual-redundant controller pair out of failover (SET NOFAILOVER) with unwritten cached data present in the write-back cache modules. Doing so will destroy data. Use the (SHOW THIS_CONTROLLER) and the (SHOW OTHER_CONTROLLER) commands to confirm that cache data has been written. Refer to Chapter 2, Section 2.1.12 for more information about write-back cache operation.

5.7.5 Failing Over

A failed or unresponsive controller in a dual-redundant configuration is disabled by its companion controller. The functioning controller sends a signal to the other controller to induce failover. The functioning controller assumes control of the storage devices that were online to the disabled controller. Maintenance can now take place on the failed controller.

Failover normally completes in 30 seconds or less (15 seconds or less for three-port controllers). If there is no outstanding drive I/O activity at the time of controller failure, failover should require substantially less than 30 seconds. If drive I/O is in progress at the time of failure, the surviving controller must reset any SCSI buses with outstanding I/O. These bus resets can require up to 5 seconds per port to complete.

Whenever you need to revive a controller that was disabled, you must enter the following command from a terminal connected to the functioning controller.

```
CLI> RESTART OTHER_CONTROLLER
```

Then initialize the controller by pressing the reset (//) button on the controller to be revived.

You may test failover by removing the program card from one of the controllers. The other controller will assume service to the dormant controller’s devices until you reinsert the program card and reinitialize/restart the controller.

5.7.6 Transparent Controller Failover Resulting from a Fault

Once redundant controllers establish a communications link in a normally operating SWXRC-04 controller storage subsystem, they maintain the link with periodic status checks. If one controller fails, the other controller senses the situation and begins the failover operation.

The sensing controller's first action is to assert the KILL signal, locking the failing controller out of any further subsystem control. Once a controller asserts this signal, it cannot be disabled by the companion controller. In the event that both SWXRC-04 controllers assert the KILL signal (as each senses a bad UART connection, for example), the first controller to assert the signal gains control of the subsystem.

In normal operation, each controller keeps a record of the subsystem configuration of the other controller. When a failure occurs, the surviving controller can then take over control of its companion's cache module and storage devices.

NOTE

RAIDsets do not fail over for SWXRC-04 dual-redundant controller configurations individually: all RAIDsets failover together as the surviving controller takes over the relevant SCSI target IDs and associated LUNs.

Note that the failover operation takes place at the controller level, in a manner completely transparent to the host. As long as the redundant controllers reside on the same SCSI bus, the host is not required to be actively involved. The transparent failover process is implemented in the SWXRC-04 controller as a simulated power fail situation.

The failover action appears to the host as a power failure, in which there is a complete subsystem reinitialization, and one controller begins servicing all of the targets of both of the original controllers. The simulated power fail situation consumes a large amount of time with respect to normal host activity, and any outstanding host requests to the failed controller normally time out. When the host reinitiates such requests, the surviving controller services them. The time out of the host's requests is the only indication to the host that a fault of some kind has occurred.

The surviving controller continues to monitor the status of the failed controller until it is restarted or replaced by the user. Once the failed controller is replaced and its replacement reestablishes communication with the surviving controller, the failback operation begins, returning subsystem control to the replacement controller for its SCSI targets.

5.7.7 Transparent Controller Failover Resulting from Operator Action

You can initiate a failover operation by running the C_SWAP local program from the CLI interface, for the purposes of swapping a controller or cache module. The C_SWAP program forces the controller on which it is executed to assume control over the entire storage subsystem.

5.7.8 Exiting Failover

You should rarely force a dual-redundant controller pair out of the failover configuration. However, the circumstances in which you may safely do this are described below.

5.7.8.1 Before Failover Occurs

If you have two normally operating dual-redundant controllers set for failover, and you want to remove one controller for use somewhere else, shut down one controller (see Chapter 9). Shutting down one controller forces failover to occur, which preserves access to your entire configuration through the surviving controller.

5.7.8.2 After Failover Occurs

After one controller in your dual-redundant pair fails or is shut down, the surviving controller services your entire configuration. To take the surviving controller out of the failover configuration, enter the following command:

```
CLI> SET NOFAILOVER
```

You must consider the following before entering the SET NOFAILOVER command or removing one controller:

- If the surviving controller and its cache module are functioning normally, you can SET NOFAILOVER without special preparation.
- If the surviving controller is running with a low write-back cache battery (enter SHOW controller to check the battery), replace/recharge the battery before entering SET NOFAILOVER and before removing the failed controller's cache module.
- Do not take a dual-redundant controller pair out of failover (SET NOFAILOVER) with unwritten cache data present in the write-back cache module. Doing so destroys data. Use the SHOW THIS_CONTROLLER command to confirm that cache data has been written.

Entering SET NOFAILOVER removes the controller from the failover configuration (as well as the other controller, if it is reachable). You may then make the configuration changes under the surviving controller.

5.7.9 Resolving a Configuration Mismatch after a Hardware Mismatch

The controller configuration is stored in nonvolatile memory on the controller module. Therefore, when one controller module is substituted for another (as might occur when replacing a failed controller), the new module's configuration contents will not match that of the surviving running controller. When the replaced controller is restarted, the configuration discrepancy is detected and a suitable error message is reported.

When a configuration mismatch is detected, the restarted controller will not recognize or access any devices. It is essentially disabled until the configurations are made the same.

To resolve a configuration mismatch, enter the following command on the surviving controller (configuration source) with the known good configuration:

```
CLI> SET FAILOVER COPY=THIS_CONTROLLER
```

This will copy the known good configuration to the newly restarted other controller.

Configuration mismatch also can occur when a subsystem that has never been configured is started for the first time. The two controller nonvolatile memories can contain different configurations left over from their last use.

To resolve a mismatch when neither controller contains a known good configuration, examine the two available configurations, chose one, and issue the SET FAILOVER COPY=THIS_CONTROLLER command to that controller to replicate the configuration on the other (the maintenance terminal must be plugged into the controller with the good configuration information).

5.7.10 Failover Setup Mismatch

During failover mismatch, one controller functions while the second controller does not recognize any devices. Although it is rare, a failover mismatch may occur during the following scenarios:

- If the controllers initialize at exactly the same time, one controller could be set for failover while the other is not.
- If one controller is running (operating normally) while the second controller is initialized, mismatch could occur. For example, this may happen after one controller underwent maintenance.

To correct a failover mismatch, stop all host processes on the devices for both controllers. Then enter the following commands to determine which controller has the desired, good configuration information:

```
CLI> SHOW UNITS
CLI> SHOW STORAGESETS
CLI> SHOW DEVICES
```

After deciding on one of the two configurations, use the SET FAILOVER command to copy the good information from one controller to the other.

5.7.11 Transparent Failback Recovering from a Fault

When the failed controller returns to service after being replaced and/or reinitialized, both controllers in the redundant pair stage a simulated power fail situation. As each controller reinitializes, it checks the least significant bit (LSB) of its own target ID. If the LSB is 0, the controller takes over the lowest of the available host bus target ID (or IDs, for multiple SCSI target configurations.) If the LSB is 1, the controller takes over the higher of the available host bus target ID (or IDs, for multiple target configurations.) At the conclusion of the power fail simulation, each controller services its original targets.

You can initiate a failover operation by running the C_SWAP local program from the CLI interface, for the purposes of swapping a controller or cache module. The C_SWAP program forces the controller on which it is executed to assume control over the entire storage subsystem.

5.8 Moving Devices Between Controllers

The moving of devices from one controller to another is supported under the following conditions:

- **Nontransportable devices**

  Under normal operation, the controller makes a small portion of a disk inaccessible to the host and uses this area to store metadata. Metadata improves error detection and media defect management. Devices utilizing metadata are called nontransportable. Initializing a device that is set as **nontransportable** places/resets metadata on the device.

  When bringing other SWXRC-04 controller (nontransportable) devices to an SWXRC-04 controller subsystem, simply add the device to your configuration using the ADD command. Do not initialize the device or the forced error information on the device is destroyed/reset.

  When adding devices, the controller firmware verifies that metadata is present. If in doubt, try to add the device so that the controller checks for metadata. If an error stating that there is no metadata occurs, initialize the device before adding it.

  A nontransportable device is interchangeable with another SWXRC-04 controller subsystem.

- Transportable devices

  A **transportable** feature is provided for the transfer of devices between non-SWXRC-04 controller systems and SWXRC-04 controller arrays. Transportable devices do not have metadata on them, and initializing a device after setting it as transportable destroys metadata (if any) on the device.

  Before moving devices from an SWXRC-04 controller subsystem to a non-SWXRC-04 controller system, delete the unit associated with the device and set the device as transportable. Then, initialize the device to remove any metadata.

When bringing non-SWXRC-04 controller devices to an SWXRC-04 controller subsystem, initialize the device after setting it transportable, then copy the data on the device to another, nontransportable, unit. Then, reinitialize the device after setting it nontransportable (thereby putting metadata on the device). You *must* initialize these devices because they may contain intact metadata blocks, which can "fool" the controller into attempting to run with the device.

CAUTION

Do not keep any device set as transportable on an SWXRC-04 controller subsystem. Doing so sacrifices forced error support on all units attached to the device.

A transportable device is interchangeable with any SCSI interface that does not utilize the device metadata (for example, a UNIX system or a PC). Transportable devices may not have write-back caching enabled, may not be members of a storageset, and do not support forced error.

NOTE

Be careful not to confuse the terms "transportable" and "nontransportable" with the qualifiers TRANSPORTABLE and NOTRANSPORTABLE. See the ADD or SET *unit-number* commands in Appendix B for more information on these qualifiers.

Transportable/nontransportable device support is summarized in Table 5-1.

Table 5–1 Transportable and Nontransportable Devices

Media Format	UNIX System	SWXRC-04 Controller
Transportable	Yes	Yes
Nontransportable	No	Yes

5.9 Moving Devices Under the Same Controller

CAUTIONS

Do not use a controller failure situation as an opportunity to move devices or otherwise reconfigure your subsystem. Doing so prevents the controller from communicating with its units once the fault is corrected.

If you lose track of the storageset members at any point during this procedure, attempt to restore the storageset by guessing where its members are installed. There is currently no way to retrace your steps using the controller or SWXRC-04 operating firmware.

You can physically relocate some or all of a storageset's member devices according to the following procedure. (This procedure also applies to reduced RAIDsets, but you must remember to add the RAIDset as REDUCED when you re-add it.)

1. Make note of all devices comprising the storageset. Digital recommends marking them after using the CLI> LOCATE command to find all storageset members.
2. Delete the unit associated with the storageset.
3. Delete the storageset.
4. Delete the device(s) to be moved.
5. Move the device(s) to the new port/target/LUN (PTL) location.
6. Add the device(s) using the new PTL location.
7. Re-add the storageset. Make sure you create it from the exact, original set of devices.

CAUTION

Do not initialize the storageset. Doing so destroys its data.

8. Re-add the unit.

The following example shows the unit "D100" made of stripeset "STRIPE0." "STRIPE0" has member disks at PTLs 200 and 210. The member at PTL 210 can be relocated to PTL 300 as follows:

```
CLI> DELETE D100
CLI> DELETE STRIPE0
CLI> DELETE DISK210
 (Move the disk to PTL 300.)
CLI> ADD DISK DISK300 3 0 0
CLI> ADD STRIPESET STRIPE0 DISK200 DISK300
CLI> ADD UNIT D100 STRIPE0
```

6

Working with RAID Arrays

This chapter describes how to work with RAID arrays.

6.1 RAID Overview

> **NOTE**
>
> For discussions in this manual, the term stripesets refers to RAID level 0, the term mirrorsets refers to RAID level 1, and the term RAIDsets refers to RAID level 5 (with RAID level 3 features).

The disk striping, mirroring, and RAID level 5 facilities of the SWXRC-04 controller provide you with a variety of options for varying the cost, performance, and data availability characteristics of attached disk storage devices. The disk striping facility (RAID 0) included with the firmware provides high I/O performance for applications requiring either high I/O request rates or high data transfer rates. The mirroring facility (RAID 1) provides maximum availability, protecting data against disk failure by replicating all data stored. The RAID facility in conjunction with the write-back cache module, combines elements of RAID level 5 and RAID level 3 technology to provide improved availability over striping (with reduced performance), but less availability than mirroring (at lower cost).

To use these RAID facilities effectively, you need to make some configuration decisions. Study the controller's CLI commands listed in this chapter and in Appendix B to configure your stripesets, mirrorsets, and RAIDsets.

In the controller firmware, 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 disks 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 failure (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 disks. **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 disks in the RAIDset to regenerate the missing data.

For detailed information about RAID technology, refer to *The RAIDBOOK – A Source for RAID Technology* published by The RAID Advisory Board, St. Peter, MN.

6.1.1 RAID Level 0

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, A, B, C, and D, for example, the first chunk is written on disk A, the second on disk B, the third on disk C, the fourth on disk D, the fifth on disk A, and so on.

CAUTION

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

The system administrator sets the chunk size based upon application requirements. If the chunk size is set to be relatively large relative 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 chunk size 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 chunk size, RAID level 0 improves either data transfer rate or I/O request rate.

6.1.2 RAID Level 0 & 1

The controller firmware supports RAID level 0 & 1, which is described as stripesets whose members are mirrorsets – called striped mirrorsets.

6.1.3 RAID Level 1

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 and connecting hardware are duplicated). For some I/O intensive applications, a RAID level 1 array can improve performance significantly over a single disk.

The controller 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 SWXRC-04 controller disk or stripeset unit, and then disassociate 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
- Disassociating the clones from the primary units
- Reenabling the application with only the primary units

Like the RAID level 5 option, the mirroring option is integrated with the controllers' dual-redundant failover capability, protecting data access from controller failure as well as disk failure. Also, like the RAID level 5 option, write performance can be enhanced by enabling write-back caching of user data for each mirrorset.

6.1.4 RAID Level 3

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 SWXRC-04 array controller 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 (in write-back 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. RAID level 3 performance is accomplished by writing to all RAIDset members in parallel with stripe-aligned writes of stripe size or larger.

NOTE

Chunk size is set with the CLI INITIALIZE command. For more information, refer to Appendix B.

6.1.5 RAID Level 5

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 repaired.

6.2 RAID Array Terminology

The following are terms are used in describing controller RAID array technology:

- COPYING member – In a mirrorset, a copying member is a member that was introduced to the mirrorset after it has been in use for some amount of time. None of the blocks can be guaranteed to be the same as other members (in contrast to NORMALIZING, where all blocks written since creation are known to be the same), therefore it is made the same by copying all the data from a NORMAL member. When all of the blocks on the COPYING member are the same as those on the NORMAL member, the COPYING member becomes a NORMAL member. Until it becomes NORMAL, the COPYING member contains undefined data and is not useful for any purpose.
- Chunksize – The number of blocks written to one RAIDset or stripeset member before data is written to the next RAIDset or stripeset member. For stripesets, the maximum chunksize is 32,768 blocks.
- Container – Any entity capable of storing data, whether it is a physical device or a group of physical devices. A disk, a stripeset, a mirrorset, and a RAIDset are examples of containers.
- Failedset – A group of disk drives that have been removed from RAIDsets or mirrorsets due to a failure or manual removal. Disk drives in the failedset should be considered potentially defective and should be tested and repaired before being placed in the spareset pool or back in their original locations.
- Mirrorset – A virtual disk drive consisting of multiple physical disk drives, each of which contains a complete and independent copy of the entire virtual disk's data.
- Nominal membership – The desired number of mirrorset members when the mirrorset is fully populated with active devices. If a member is removed from a mirrorset, the actual number of members may fall below the “nominal” target.
- Normalizing state – the state that occurs within the mirrorset from either of two conditions:
  - During initialization after the mirrorset is first created by CLI
  - After a cache data lost (CDL) situation has been cleared by the user. Note that all data written prior to clearing CDL is undefined. All data written after clearing CDL is consistent across the mirrorset

  As a result of the Normalizing state occurring, mirrorset members become either NORMAL members or NORMALIZING members:

  - NORMAL member – A mirrorset member whose entire contents is guaranteed to be the same as all other NORMAL members. All NORMAL members are exactly equivalent.
  - NORMALIZING member – A mirrorset member whose contents is the same as all other NORMAL and NORMALIZING members for data that has been written since the mirrorset was created or since lost cache data was cleared. Data that has never been written may differ among NORMALIZING members.

When a mirrorset is brought online, if no NORMAL members are present, one of the NORMALIZING members is chosen and set to NORMAL. This NORMAL member becomes the source of data for making all remaining NORMALIZING members the same. The mirrorset is made completely NORMAL when all of the source data has been copied.

All mirrorset members enter NORMALIZING state together, and at the proper time, one is chosen as the NORMAL member. If this NORMAL member then fails out of the mirrorset, then any of the other NORMALIZING members may be chosen as the new NORMAL member.

- Parity – Any kind of checksum that allows the regeneration of unretrievable data. Parity is typically combined with data stored in positionally corresponding blocks of other disks in the RAIDset to regenerate the missing data.
- RAIDset – A virtual disk drive with its physical data spread across multiple physical disks. A RAIDset contains parity data to be used to regenerate data in the event that one member fails.
- RAIDset states
  - Normal state – All members are present and all data is redundant.
  - Reduced state – A failed RAIDset member has been detected and removed from the RAIDset.
  - Reconstructing state – All members are present and redundancy is being restored.
    - Reconstruct types (process of restoring redundancy to the RAIDset)

      There are two different types of reconstruct: one that takes place when a unit is created from a RAIDset; the other takes place when a failed RAIDset member is replaced. Each kind indicates a different error recovery operation/choice that the controller will make.

      - Initial reconstruct – Establishes initial redundancy following an ADD RAIDset...ADD UNIT command sequence. Note that all data written by the host is immediately fully redundant.

        When a RAIDset is initialized using the INITIALIZE command, the controller does not take the lengthy period of time to make all the parity blocks consistent with the data. Instead, the controller marks all the parity blocks as bad and starts a reconstruct. The reconstruct recalculates and rewrites the parity blocks and marks them as good. This process allows the RAIDset to be used immediately. All new data written to the RAIDset is immediately fully redundant.

      - Reconstructing a replaced member – Regenerates the data for that member and restores redundancy.

        Reconstruct means to restore redundancy. This could be either recalculating the parity or recalculating a user data block, using the remaining blocks.

When a reduced RAIDset has a member added back to it, all the blocks on the replacement member are marked as bad (parity and user data blocks), and a reconstruct scan is started. The reconstruct recalculates the parity blocks on the new member and recalculates the user data blocks on the new member, writes the blocks, and marks the blocks good.

NOTE

If a second RAIDset member fails during a reconstruct operation, the RAIDset becomes inoperative.

To determine which type of reconstruct is taking place, enter the SHOW RAIDsets command. If more than one reconstructing member is identified, the controller is performing an initial reconstruct.

- Redundancy – A RAIDset is considered to be redundant when user data is recorded directly to one member, and all of the other members and associated parity also are recorded. If a member is missing from the RAIDset, its data can be regenerated as needed, but the RAIDset is no longer redundant until the missing member is replaced and reconstructed.
- Regenerate – The process of calculating missing data from the redundant data.
- Replacement policy – The firmware controlled method by which a spare disk is selected to replace a disk that has failed in a RAIDset or mirrorset. Your replacement policy choices are BEST_FIT, BEST_PERFORMANCE, or NOPOLICY. Refer to the ADD RAIDSET, ADD MIRRORSET, and SET *raidset-container-name* CLI commands in Appendix B.
- Spareset – A pool of disk devices available to the controller to replace failed RAIDset or mirrorset members.
- Stripe – The data and parity from the associated chunks of each member of the RAIDset.
- Stripe size – The capacity determined by *n*-1 times the chunksize. (*n* is the number of RAIDset members).
- Stripeset – A virtual disk drive with its data spread across multiple physical disks. (Currently, the stripeset chunksize maximum is 32,768 blocks.)

6.3 RAIDset and Mirrorset Rules and Important Information

The following list gives rules to remember about RAIDsets and mirrorsets:

- You must always have a write-back cache module when creating RAID level 5 RAIDsets or mirrorsets.

To utilize write-back caching, RAID level 5, and mirroring, you must purchase licenses and enter the valid license key(s). Using these facilities without entering the valid license key(s) causes an hourly error message to be displayed at the terminal. Entering the valid license key(s) stops the error messages.

NOTE

You may activate write-back caching, RAID level 5, and mirroring via the firmware licensing system (FLS) utility.

- NOWRITEBACK_CACHE (write-through caching) is automatically set for units created from RAIDsets or mirrorsets. To increase the unit's performance, switch to WRITEBACK_CACHE.
- RAIDsets may be made up of 3 to 14 members. Mirrorsets may be from 2 to 6 members.
- RAIDsets and mirrorsets in a dual-redundant configuration flush cache and failover to the companion controller if the write-back cache module battery for *one* controller in the pair has a low charge and CACHE_POLICY=B is set.
- RAIDsets and mirrorsets will go inoperative (and write-protected) if *both* write-back cache modules' batteries fail, or if a single cache's batteries fail in a nonredundant controller. Any stripesets will flush cache data and become write-through (cache module acts as read cache) instead of write-back. RAIDsets and mirrorsets become operational when the batteries become fully charged again.

NOTE

When the power for the write-back cache module batteries is too low, a console message is displayed. You can check the status of the batteries at any time by entering the CLI SHOW THIS_CONTROLLER command (or SHOW OTHER_CONTROLLER as appropriate). Depending on which cache policy you choose, your RAIDsets and mirrorsets will either go inoperative (CACHE_POLICY=A) or remain available in write-through mode (CACHE_POLICY=B) when the batteries go low.

- Your RAIDsets can achieve the high performance characteristics of RAID level 3 provided you set your RAIDset chunksize to the minimum value (16) and your application calls for large sequential I/O operations.
- RAIDsets and mirrorsets can contain disks of different sizes, but the disk space used by the storageset is limited to the maximum size of the smallest disk in the RAIDset.
- Place RAIDset and mirrorset members on different ports. This keeps your storageset from going inoperative in the event that a single port bus failure occurs.
- You may have a combined maximum total of 20 mirrorsets and RAIDsets.
- You may have a combined maximum total of 30 storagesets (mirrorsets, RAIDsets, stripesets).
- You can have no more than 32 physical device members for a unit (practically, this applies to RAID 0 & 1 configurations).

6.4 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 configuring dual-redundant controllers for failover, make sure the target controller (the controller you are copying configuration information "to") has no cache errors or unwritten cached data. Delete any units on the target controller to verify there are no cache errors, before entering the SET FAILOVER command.
- Do not take a dual-redundant controller pair out of failover (by entering the SET NOFAILOVER command) with unwritten cached data present in the write-back cache modules. Doing so will destroy data. Enter the SHOW THIS_CONTROLLER and SHOW OTHER_CONTROLLER commands to confirm that cached data has been written.

NOTE

When the SET NOFAILOVER command is issued, the other controller shuts down.

- 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 a 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 associated 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.

6.4.1 Using CLI Commands with Write-Back Cache

A write-back cache module is required for RAID operations.

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.

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.

6.5 Planning Your RAIDsets

The following items should be considered before creating your RAIDsets:

- RAIDset size (3 to 14 members)
- RAIDset chunk size
- RAIDset replacement policy
- RAIDset reconstruction rate
- RAIDset spares
- RAIDset configurations for availability and performance
- RAIDset hardware requirement

6.5.1 Creating a RAIDset

Enter the following commands to create a RAIDset:

1. Use the ADD DISK *container-name PTL* command to add new disk drives to your configuration and name them.

   ```
   CLI> ADD DISK DISK0 1 0 0
   ```

2. Use the ADD RAIDSET *container-name container-name1 container-name2 [container-nameN]* command and set the appropriate replacement policy and reconstruct qualifiers. You are not required to set a replacement policy or reconstruct rate.

   The following is an example for using the replacement policy and reconstruct qualifiers on the same command line as the ADD RAIDSET command:

   ```
   CLI>  ADD RAIDSET RAID0 DISK100 DISK200 DISK300 POLICY=BEST_FIT
         RECONSTRUCT=NORMAL
   ```

   Where:

   - RAID0 is the name assigned to the RAIDset.
   - DISK100, DISK200, and DISK300 are the names assigned to the RAIDset members of a three-member RAIDset.
   - POLICY=BEST_FIT is the replacement policy qualifier that will be used when a RAIDset member fails. (You can choose between three different replacement policy qualifiers: POLICY=BEST_FIT, POLICY=BEST_PERFORMANCE, or NOPOLICY.)
   - RECONSTRUCT=NORMAL is the reconstruction rate qualifier that is used when a RAIDset member fails and a new member is taken from the spareset as a replacement for the failed device. You can choose between two reconstruct qualifiers: RECONSTRUCT=NORMAL (the default), or RECONSTRUCT=FAST.

3. Enter the INITIALIZE command for your RAIDset. This is the time to specify your chunk size. The metadata on the container (in this case, the RAIDset) must be initialized before a unit may be created from it. If the container's metadata cannot be found, or is incorrect, an error will be displayed and the unit will not be created.

   ```
   CLI> INITIALIZE RAID0 CHUNKSIZE=n
   ```

   Where *n* is the chunk size in blocks.

   or

   ```
   CLI> INITIALIZE RAID0 CHUNKSIZE=DEFAULT
   ```

   Where the controller determines the chunk size.

4. Enter the ADD UNIT *unit-number container-name* command to create a host accessible logical unit from the RAIDset, followed by the appropriate qualifier for cache transfer sizes, preferred path, cache access, write protection, and so forth (as described in Appendix B.

   ```
   CLI> ADD UNIT D107 RAID0
   ```

   Where:

   - D107 is the unit name.
   - RAID0 is the RAIDset name.

5. Enter the SHOW RAIDSETS command to display all of the RAIDsets known to the controller. By adding the FULL qualifier, more information concerning all of the known RAIDsets is displayed.

   To show information about a particular RAIDset, enter the SHOW *raidset-container-name* command. Where *raidset-container-name* is the name assigned to the particular RAIDset.

6. Using the ADD SPARESET command, populate the spareset pool with disk drives that closely match the geometry of the other disk drives in your subsystem.

Refer to Appendix B for descriptions and examples for choosing and using the appropriate ADD UNIT qualifiers.

6.5.2 Storageset SHOW Commands

The term storageset is used to refer to RAIDsets, stripesets, or mirrorsets. The following CLI SHOW commands are used to display the status of your RAIDsets:

Enter the following CLI command to display all RAIDsets known to the controller:

```
CLI> SHOW RAIDSETS
Name            Storageset      Uses         Used by
-----------------------------------------------------------
RAID0           raidset         DISK100      D107
                                DISK200
                                DISK300
```

Enter the following CLI command to display additional information about all RAIDsets known to the controller:

```
CLI> SHOW RAIDSETS FULL
Name            Storageset      Uses          Used by
----------------------------------------------------------
RAID0           raidset         DISK100       D107
                                DISK200
                                DISK300
     Switches:
       POLICY (for replacement) = BEST_FIT
       RECONSTRUCT (priority) = NORMAL
       CHUNKSIZE = 256 blocks
     State:
       RECONSTRUCT 3% complete
       DISK100   (member 0) is RECONSTRUCTING
       DISK300   (member 1) is RECONSTRUCTING
       DISK400   (member 2) is RECONSTRUCTING
       DISK500   (member 3) is RECONSTRUCTING
     Size: 2050353 blocks
```

Enter the following CLI command to display information about a particular RAIDset:

```
CLI> SHOW RAID0
Name            Storageset      Uses          Used by
----------------------------------------------------------
RAID0           raidset         DISK100       D107
                                DISK200
                                DISK300
     Switches:
       POLICY (for replacement) = BEST_FIT
       RECONSTRUCT (priority) = NORMAL
       CHUNKSIZE = 256 blocks
     State:
       NORMAL
       DISK100   (member 0) is NORMAL
       DISK200   (member 1) is NORMAL
       DISK300   (member 2) is NORMAL
     Size: 2050353 blocks
```

Enter the following CLI command to display all information about all stripesets known to the controller:

```
   CLI> SHOW STRIPESETS FULL
Name            Storageset      Uses         Used by
----------------------------------------------------------
STRIPE0         stripeset       DISK310      D1
                                DISK410
                                DISK510
     Switches:
       CHUNKSIZE = 256 blocks
     State:
       NORMAL
       DISK310   (member 0) is NORMAL
       DISK410   (member 1) is NORMAL
       DISK510   (member 2) is NORMAL
     Size: 25134084
```

Enter the following CLI command to display all information about all storagesets known to the controller:

```
   CLI> SHOW STORAGESETS FULL
Name            Storageset      Uses       Used by
----------------------------------------------------------
STRIPE0         stripeset       DISK110    D1
                                DISK210
                                DISK310
     Switches:
       CHUNKSIZE = 256 blocks
     State:
       NORMAL
       DISK310   (member 0) is NORMAL
       DISK410   (member 1) is NORMAL
       DISK510   (member 2) is NORMAL
     Size: 25134084

RAID0           raidset         DISK100    D107
                                DISK300
                                DISK400
     Switches:
      POLICY (for replacement) = BEST_FIT
      RECONSTRUCT (priority)  = NORMAL
      CHUNKSIZE = 256 blocks
     State:
      NORMAL
      DISK100   (member 0) is NORMAL
      DISK200   (member 1) is NORMAL
      DISK300   (member 2) is NORMAL
     Size: 16751956

SPARESET        spareset        DISK400
                                DISK610
FAILEDSET       failedset       DISK500
```

6.5.3 Adding and Deleting Spareset Members

The spareset is a pool of disk drives available to the controller to replace failing members of a RAIDset or mirrorset. The ADD SPARESET command adds disk drives to the spareset pool and initializes the metadata on the drives so they may be used for replacements into RAIDsets and mirrorsets. The DELETE SPARESET command removes disk drives from the spareset.

> **NOTE**
>
> The spareset cannot be deleted, it is always available.

Enter the following CLI commands to add a disk to the spareset:

```
CLI> ADD SPARESET disk-container-name0
```

Example:

```
CLI>   ADD SPARESET DISK400
```

Enter the following CLI commands to remove one or more disks from the spareset:

```
CLI> DELETE SPARESET disk-container-name0 [disk-
     containter-nameN]
```

Example:

```
CLI>   DELETE SPARESET DISK400
CLI>   DELETE SPARESET DISK100 DISK200 DISK300
```

Enter the following command to show the spareset:

```
CLI> SHOW SPARESET
Name            Storageset      Uses       Used by
----------------------------------------------------------

SPARESET        spareset        DISK400
                                DISK610
```

6.5.4 Showing and Deleting Failedset Members

The Failedset is a group of disk drives that were removed from RAIDsets or mirrorsets because they failed or were manually removed (via the SET *RAIDset-container-name REMOVE=disk-container-name* command. Drives in the Failedset should be considered defective. These drives must be tested and repaired before placing them back in operation.

The DELETE FAILEDSET command removes drives from the failedset so that they can be physically removed from the device shelves for testing and repair. Enter the following commands to show, and then remove, one or more disk drives from the failedset:

```
CLI> DELETE FAILEDSET DISK99
CLI> DELETE FAILEDSET DISK99 DISK88 DISK77
```

Enter the following CLI command to show a failedset:

```
   CLI> SHOW FAILEDSET

Name            Storageset      Uses       Used by
----------------------------------------------------------
FAILEDSET       failedset       DISK500
```

NOTE

The failedset cannot be deleted, it is always available.

6.5.5 Changing RAIDset Characteristics

To change certain characteristics of a RAIDset, use the SET *RAIDset-container-name* command.

When a RAIDset looses a member, a new member is automatically added to the RAIDset from the spareset pool (providing you have a replacement policy set, and an appropriate spare is in the spareset). If you specified NOPOLICY, when you created your RAIDset, or you wish to change your replacement policy, enter one of the following commands:

```
CLI> SET RAIDset-container-name POLICY=BEST_FIT
CLI> SET RAIDset-container-name POLICY=BEST_PERFORMANCE
CLI> SET RAIDset-container-name NOPOLICY
```

To change 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, enter one of the following commands:

```
CLI> SET RAIDset-container-name RECONSTRUCT=NORMAL
CLI> SET RAIDset-container-name RECONSTRUCT=FAST
```

If you need to remove a disk member from a RAIDset, enter the following command:

```
CLI> SET RAIDset-container-name REMOVE=disk-container-name
```

For example:

```
CLI> SET RAID0 REMOVE=DISK100
```

If the RAIDset is already in a reduced state when the REMOVE=qualifier is used, an error is printed and the command is rejected. If a replacement policy is specified, the replacement drive is automatically 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 or a policy is specified. The disk drive removed via the REMOVE=qualifier is automatically added to the failedset.

To manually place a disk member into a reduced RAIDset when NOPOLICY was specified, enter the following command:

```
CLI> SET RAIDset-container-name REPLACE=disk-container-
        name
```

For example:

```
CLI> SET RAID0 REPLACE=DISK520
```

Where RAID0 is the RAIDset name, and DISK520 is the replacement disk name.

The disk called DISK520 is added to the reduced RAIDset (RAID0). A reconstruct operation begins immediately on the newly added disk (as long as the reconstruct is not disabled).

> **NOTE**
>
> No other qualifiers can be used with the SET *RAIDset-container-name* command when either the REPLACE or REMOVE qualifiers are specified.

6.5.6 Deleting a RAIDset

Use the DELETE *container-name* command to delete a RAIDset. This command determines whether the container (RAIDset) is used by a unit. If the container is in use, an error is printed and the container is not deleted. If the container is not is use, it is deleted.

Enter the following command to delete a RAIDset:

```
CLI> DELETE container-name
```

For example:

```
CLI> DELETE RAID0
```

Where RAID0 is the name of the RAIDset being deleted.

6.5.7 Moving a RAIDset

You may physically relocate some or all of a RAIDset's member devices according to the following procedure:

> **CAUTION**
>
> If you lose track of the RAIDset members at any point during this procedure, you will have to attempt to restore the RAIDset by guessing where its members are installed. There is currently no way to retrace your steps using the controller or operating firmware.

To move a RAIDset you must do the following:

1. Make note of all devices comprising the RAIDset. Digital recommends marking them after using the CLI> LOCATE command to find all RAIDset members.
2. Delete the UNIT that uses the RAIDset with the DELETE *unit-number* command.
3. Delete the RAIDset with the DELETE *container-name* command.
4. Delete each disk from that RAIDset with the DELETE *container-name* command.
5. Physically remove the disks from the storage shelf.
6. Move the disks to the new port/target/LUN (PTL) location.
7. Add each disk with the ADD DISK *container-name PTL* command using the new **PTL** location.
8. Re-add the RAIDset with the ADD RAIDSET *container-name container-name1 container-name2 [container-nameN]* command. Make sure you create it from the exact, original set of drives.

CAUTION

Do *not* initialize the RAIDset or you will destroy its data.

9. Recreate the logical unit from the RAIDset with the ADD UNIT *unit-number container-name* command.

The following example shows the unit “D100” made of RAIDset “RAID99.” “RAID99” has member disks at PTLs 200, 210, and 400. The member at PTL 210 can be relocated to PTL 300 as follows:

```
CLI> DELETE D100
CLI> DELETE RAID99
CLI> DELETE DISK210
```

Move the disk to PTL 300.

```
CLI> ADD DISK DISK300 3 0 0
CLI> ADD RAIDSET RAID99 DISK200 DISK300 DISK400
CLI> ADD UNIT D100 RAID99
```

If you move a RAIDset from one controller to another and you damage one member, you must specify all of that RAIDset's members when you re-add the RAIDset to the new controller. The controller will automatically reduce the RAIDset when it discovers that one member is inoperative.

Using the REDUCED Qualifier with the ADD RAIDset Command

Only use the REDUCED qualifier (with the ADD RAIDSET command) when you want to move a RAIDset that is already reduced. For example, you have a four member RAIDset that has been reduced to a three member RAIDset on Controller A and you wish to move the RAIDset to Controller B. Physically move the three members to Controller B and enter the following command:

```
CLI> ADD RAIDSET container-name container-name1 container-
     name2 container-name3 REDUCED
```

For example:

```
CLI> ADD RAIDSET RAID0 DISK100 DISK300 DISK400 REDUCED
```

6.6 Adding a Stripeset (RAID Level 0)

Use the ADD STRIPESET *container-name container-name1 container-name2* command to add a stripeset and to name that stripeset. This command must be used when a new stripeset is added to a controller's configuration. A stripeset may contain from 2 to 14 members. To create a stripeset, add the individual disks, add the stripeset and name it, initialize the stripeset, and then create and name a unit from the stripeset as shown in the following example:

```
CLI> ADD DISK DISK99 1 0 0
CLI> ADD DISK DISK88 2 0 0
CLI> ADD DISK DISK77 3 0 0
CLI> ADD STRIPESET STRIPE0 DISK99 DISK88 DISK77
CLI> INITIALIZE STRIPE0
CLI> ADD UNIT D0 STRIPE0
```

6.6.1 Moving a Stripeset or Stripeset Member

You may physically relocate some or all of a stripeset's member devices. However, if you lose track of the stripeset members at any point during the relocation, you will have to attempt to restore the stripeset by guessing where its members are installed. There is currently no way to retrace your steps using the controller or operating firmware.

Use the same procedure as described in Section 6.5.7 to move a stripeset or stripeset member.

6.6.2 Showing Stripesets

The SHOW STRIPESET command displays all the stripesets known by the controller. The SHOW STRIPESET FULL gives more information about all stripesets known to the controller. By entering the SHOW *stripeset-container-name* command, you are given specific information about a particular stripeset.

```
CLI> SHOW STRIPESET
CLI> SHOW STRIPESET FULL
CLI> SHOW stripeset-container-name
CLI> SHOW UNITS FULL
```

6.7 Planning Your Mirrorsets

- The following items should be considered before creating your mirrorsets:
- Mirrorset size (1 to 6 members)
- Mirrorset replacement policy
- Mirrorset spares
- Mirrorset hardware requirements – write-back cache module

6.8 Using Mirrorsets to Obtain Snapshot Copies of Data

A mirrorset is a storageset consisting of multiple devices, each containing an identical copy of the same data. Because individual devices can be added and removed from a mirrorset while it is in use (with minimal impact on the mirrorset), and because removed members contain a complete copy of all the data on the mirrorset at the time of removal, mirroring is used in some system management situations to obtain snapshot copies of data.

The general strategy used in these situations contains the following steps:

1. During normal, steady-state operation, run the unit with the number of drives appropriate for the availability required.
2. When the time comes to take a snapshot of the data, activate mirroring on the unit in question (if it is not already active), and add a physical disk into the mirrorset. As with the addition of any new mirrorset member, the mirroring facility copies the data from the existing members to the new "snapshot" member.
3. When the copy to the snapshot member is complete, extract the snapshot member from the mirrorset. In some situations, the application is quiesced so that there is no activity to the mirrorset when the snapshot member is extracted; other situations are insensitive to when the snapshot member is extracted.
4. If mirroring is not the normal steady-state mode for the unit, deactivate mirroring (UNMIRROR) on the unit that was copied.
5. Set the snapshot copy aside for use as a ready backup. Alternatively, the snapshot copy can be mounted and a conventional backup utility run against it to backup the contents to tape. This latter approach allows the backup utility to be run at locations and times more convenient to system operation than the traditional fixed backup window.

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. But for some systems, mirror snapshots are a useful technique.

The controller contains the following mirrorset commands to ease the task of using mirroring to obtain data snapshots:

The MIRROR and UNMIRROR commands allow a specific disk to be converted from a disk container to a mirrorset and back again to a simple disk container, all while the disk is in use. This means that units that will be snapshot do not have to be configured permanently as mirrorsets; mirroring can be activated and deactivated on the units only for the time it takes to obtain the snapshot.

The REDUCE command allows multiple disks to be simultaneously removed from multiple mirrorsets as a single, synchronized operation. This permits snapshot copies of stripesets and striped mirrorsets in a way that retains the integrity of the entire stripeset.

The NODESTROY option on the INITIALIZE command permits the snapshot copy to be configured and added as a unit (distinct from the original) without destroying any of the data.

6.9 Mirrorset Considerations for Snapshot Copies

If you want to use mirroring to create a snapshot copy of data, consider the following:

1. The only useful snapshot is a snapshot of all the data on a unit. This is because the host sees the unit as a single integrated storage space, and the host uses this storage space without regard to any physical boundaries in the underlying disks.

   This means that if the unit is layered on one or more storagesets, *all* devices in the storageset must be snapshot at the same time.

   Thus, if the unit is layered on a simple disk container, then the snapshot need only involve one disk. Likewise, if the unit is layered on a simple mirrorset, the snapshot need only involve one disk (because for mirrorsets, all members are equivalent). For stripesets and striped mirrorsets, however, multiple snapshot disks are involved, one for each member of the stripeset.

   To preserve the host view of the data, when you configure the snapshot unit to gain access to the snapshot data, you must carefully configure the snapshot in a manner identical to the configuration (at the time the snapshot member was removed from the mirrorset). Since all snapshot disks will have been mirrorset members at the time of removal, they must be defined as mirrorset members when used in the snapshot unit as well. Therefore:

   - A snapshot of a single disk unit must be configured as a single-member mirrorset
   - A snapshot of a mirrorset unit must be configured as a single-member mirrorset
   - A snapshot of a stripeset must be configured as a stripeset of single-member mirrorsets
   - A snapshot of striped mirrorsets must be configured as a stripeset of single-member mirrorsets

2. See the CLONE utility discussion in Chapter 8 for examples of this consideration in practice. Mirroring snapshots cannot be used for RAIDset units. They can only be used to snapshot the data in disk, simple mirrorset, simple stripeset, or striped mirrorset units.

3. The bulk of the overhead in making a snapshot is consumed during the copy phase. Once the snapshot member copy is complete (it is in the NORMAL state), it is kept current with all other members from then on. Thus, the extraction of the snapshot from the mirrorset can be done with similar results at any time after the copy is complete. Because the extraction operation is quick, it can be synchronized manually with host activity in any convenient fashion.

4. The extraction of the snapshot member is done with the REDUCE command. The copy must be done before the REDUCE will be honored. If multiple snapshot members are involved (as they will be if you are copying the data on a stripeset or striped mirrorset), then all copies on all members must be complete.
5. Snapshot disks must be the same size or larger than the disk members they will be used to copy.
6. 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.
7. Any unit created by CLONE has a mirrorset level in the configuration hierarchy, even if the original unit did not have mirroring. The target disk of a single disk unit is mirrored then added and initialized by CLONE. Because the target disk is initialized as a mirrorset member, CLONE makes it into a single-member mirrorset to preserve the metadata before adding it as a unit.
8. The CLONE utility is a handy means of automating snapshot copies if you don't want to manage the details yourself

6.10 Steps for Creating a Mirrorset

This section describes the steps necessary to capture a snapshot copy of the data on a single disk container. These steps can be accomplished automatically by the CLONE utility. Following are the specific steps to making a snapshot copy using the controller mirroring facility:

1. Identify the unit to be copied.
2. Use SHOW commands to determine the individual disk devices that make up the unit and their sizes.

```
CLI> SHOW DEVICES
CLI> SHOW STORAGESETS FULL
```

3. Identify the unused disks that you will use as the snapshot targets.
4. If the unit is a simple disk or simple stripeset unit, use the MIRROR command to convert each disk in the unit to a one-member mirrorset. If the unit is already layered on a mirrorset or striped mirrorset, this step is not necessary. If the unit being snapshot is a simple stripeset, you will create a distinct mirrorset for each member.

```
CLI> MIRROR disk-device-name container-name
```

5. The unit should now be layered entirely on mirrorsets. Use SHOW commands to satisfy yourself that this is the case.
6. For each mirrorset in the unit, set the replacement policy to NOPOLICY. This will provide more precise control over which snapshot targets are used for each disk.

```
CLI> SET mirrorset-container-name NOPOLICY
```

7. For each mirrorset in the unit, use the SET MEMBERSHIP command to increase the nominal membership of the mirrorset by one. This will provide the member slot for the snapshot member.

```
CLI> SET mirrorset-container-name MEMBERSHIP=n
```

8. For each mirrorset in the unit, use the SET REPLACE=command to add the target snapshot disk you desire into that mirrorset. At this point, the copy to the snapshot disk will begin.

```
CLI> SET mirrorset-container-name REPLACE=disk-device-
     name
```

9. Monitor all the mirrorsets for copy completion. When all members are in normal state, the extraction can be done.

```
CLI> SHOW THIS_CONTROLLER
CLI> SHOW OTHER_CONTROLLER
```

10. Extract all snapshot members from all the mirrorsets in the unit with a single REDUCE command. Name each of the snapshot disks on the same line; all will be removed at once.

```
CLI> REDUCE disk-device-name1 disk-device-name2
```

11. For each device that you issued a MIRROR command, issue an UNMIRROR command to eliminate the temporary mirrorset. If the unit is layered on permanent mirrorsets, do not perform this command.

```
CLI> UNMIRROR disk-device-name1
```

12. For each permanent mirrorset in the unit, use the SET MEMBERSHIP command to reduce the nominal membership back to where it was when you started.

```
CLI> SET mirrorset-container-name MEMBERSHIP=n
```

13. For each permanent mirrorset, use the SET POLICY=command to restore the replacement policy you want during normal operation.

```
CLI>  SET mirrorset-container-name POLICY=BEST_FIT
or
CLI>  SET mirrorset-container-name POLICY=BEST_PERFORMANCE
```

6.10.1 Configuring Host Units into Mirrorsets

Now that you have the snapshot copies in the form of the individual snapshot disks, the following steps are necessary to configure usable host units from those disks:

1. If the snapshot is of a striped mirrorset, for each of the snapshot disks, create a one-member mirrorset consisting solely of the snapshot disk. For example:

```
CLI> ADD MIRROR  mirrorset-name-1 snapshot-disk-name-1
CLI> ADD MIRROR  mirrorset-name-2 snapshot-disk-name-2
```

2. Then create a stripeset consisting of the mirrorsets just defined:

```
CLI> ADD STRIPESET stripeset-name mirrorset-name-1
     mirrorset-name-2
```

If the snapshot is of a stripeset, create a stripeset consisting of each of the constituent snapshot disks:

```
CLI> ADD STRIPESET stripeset-name snapshot-disk-name-1
     snapshot-disk-name-2
```

If the snapshot is of a simple disk or a simple mirrorset, the snapshot disk is used as a simple disk device.

2. Initialize the snapshot stripeset or device using the NODESTROY option. Only initialize the *top* storageset on which you are going to add the unit. If you copied a stripeset or striped mirrorset unit, initialize the stripeset created in step 1 above. If you copied a simple disk or mirrorset unit, initialize the simple disk unit that resulted.

   If you are initializing a stripeset or striped mirrorset, specify the same chunksize as that on the original stripeset or striped mirrorset.

   ```
   CLI> INITIALIZE container-name NODESTROY CHUNKSIZE=n
   ```

3. Add the host-visible unit with an ADD UNIT command.

6.11 Mirrorset Command Overview

The sections that follow describe mirrorset procedures, commands, and qualifiers.

6.11.1 Creating a Mirrorset

Enter the following commands to create a mirrorset:

1. Add the disks to be used for the mirrorset (unless they have already been added):

   ```
   CLI> ADD DISK DISK120 1 2 0
   CLI> ADD DISK DISK220 2 2 0
   CLI> ADD DISK DISK320 3 2 0
   ```

For the best performance and data protection, your mirrorset members should reside on different ports.

2. To create your mirrorset:

   ```
   CLI> ADD MIRRORSET MIRR1 DISK120 DISK220 DISK320
   CLI> INITIALIZE MIRR1
   CLI> ADD UNIT DO MIRR1
   ```

3. To display all of the information about your newly created mirrorset:

   ```
   CLI> SHOW MIRR1
   ```

4. To display all of the information about all of the mirrorsets known to your controller:

   ```
   CLI> SHOW MIRRORSETS FULL
   ```

6.11.2 Mirrorset SET Commands and Qualifiers

Use the SET *mirrorset-container-name* command and one of its qualifiers to change the characteristics of a mirrorset as listed:

```
CLI> SET MIRR1 MEMBERSHIP=number-of-members
CLI> SET MIRR1 REPLACE=disk-device-name
CLI> SET MIRR1 REMOVE=disk-device-name
CLI> SET MIRR1 POLICY=policy-type
CLI> SET MIRR1 READ_SOURCE=read-source
CLI> SET MIRR1 COPY=copy_speed
```

The following are the SET *mirrorset-container-name* qualifiers and descriptions for using them:

- MEMBERSHIP=*number-of-members* – Allows you to increase or decrease the nominal membership of a mirrorset to the number you specify in *number-of-members*. If the mirrorset membership is increased, and automatic sparing is turned on, the mirrorset automatically brings in spares until the new number of members is attained, or until no more suitable spares are available. Membership cannot be decrease below the number of active members. Remove members first, then reduce membership.
- REPLACE=*disk-device-name* – Allows you to add mirrorset members to an existing mirrorset. There are two conditions for using this qualifier:
  - The replacement policy must be set to NOPOLICY.
  - The mirrorset must be missing at least one member.

  If these two conditions are met, the physical device is added to the mirrorset you named. The nominal number of members does not change.
- REMOVE=*disk-device-name* – Allows you to remove a member from an existing mirrorset. If the physical disk you specify is not a member of the mirrorset you specified, or if the mirrorset will not have a NORMAL or NORMALIZING member after the disk is removed, an error is reported and no action is taken. If the disk removal is successful, the removed disk is added to the failedset, and a new disk member is auto-spared into the mirrorset (if applicable). The nominal number of members in the mirrorset does not change. If for some reason the auto-sparing does not take place, the mirrorset automatically adds the spare when an acceptable spare becomes available or when the replacement policy changes.
- REDUCE *disk-device-name1 [disk-device-nameN]* – Allows you to remove members from an existing mirrorset. Refer to Section 6.11.4 in this chapter.
- POLICY=*policy-type* – Allows you to set the desired automatic replacement policy for disks that fail out of a mirrorset.
  - BEST_FIT gives highest priority to finding a replacement device within the spareset that closely matches the sizes of the remaining members of the mirrorset. If several spareset members are the same size, the best performance algorithm is used to determine the best device to add to the mirrorset.

  - BEST_PERFORMANCE (default) gives the highest priority to finding a replacement device that is on a different port than any of the current mirrorset members. If several spareset members are on different ports, then the best fit algorithm is used to determine the best device to add to the mirrorset.
  - If there are no spareset members that are at least the size of the mirrorset, then the mirrorset is left with the reduced number of members.
  - NOPOLICY – Allows you to turn off the autosparing capability. Allows you to remove a failing device from a mirrorset without selecting a replacement. This causes the mirrorset to run in a reduced state until a BEST_FIT or BEST_PERFORMANCE policy is selected, or a member is manually replaced in the mirrorset. If the BEST_FIT or BEST_PERFORMANCE policy is set and an acceptable spare is made available, the spare is automatically added to the mirrorset.
- READ_SOURCE=read-source – Allows you to control the read algorithm for the specified mirrorset Choose from the following:
  - 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 (default) – The NORMAL mirrorset member with the least busy work queue is the target of the read.
  - Preferred member – The specified member is used for all reads.
- COPY=*copy_speed* – Allows you to specify the speed at which mirrorset copies are performed. The copy_speed choices are as follows:
  - NORMAL copies the mirrorset using minimal resources.
  - FAST copies the mirrorset using multiple resources, thus speeding up the copy, but slowing down normal I/O.

6.11.3 Mirrorset SHOW Commands

SHOW MIRRORSETS – Displays all configured mirrorsets.

SHOW MIRRORSETS FULL – Additional information will be displayed after each mirrorset.

SHOW *mirrorset-container-name* – Displays the same information as SHOW MIRRORSETS FULL, except that it only displays information for the mirrorset specified by *mirrorset-container-name.*

6.11.4 REDUCE *disk-device-name1* Command

The REDUCE *disk-device-name1 [disk-device-nameN]* command is use for removing multiple members from mirrorsets that make up a stripeset. For a consistent copy of the stripeset, all members MUST be removed at the same time (not individually). The REDUCE command is similar to the SET *mirrorset-container-name* REMOVE=*device-name* command except that the nominal number of members in the mirrorset are decrease by the number of members removed, and the devices are not placed in the failedset.

The disk devices to be removed need not be members of the same mirrorset, but the devices MUST be part of the same unit. No autosparing occurs and each mirrorset membership is set to the new reduced number of members. For each mirrorset that you reduce, at least one remaining NORMAL member must remain in the mirrorset. If not, none of the specified mirrorsets are reduced.

6.11.5 MIRROR *disk-device-name container-name* Command

Allows you to convert a physical device to a one member mirrorset.

```
CLI> MIRROR disk-device-name mirrorset-container-name
```

For example:

```
CLI> MIRROR DISK100 MR1
```

6.11.6 UNMIRROR *disk_device-name*

Allows you to convert a one member mirrorset back to a physical device.

```
CLI> UNMIRROR disk-device-name
```

Refer to Chapter 8 for mirrorset examples performed by the CLONE utility

6.12 RAIDset and Mirrorset Availability, Performance, and Cost

RAIDset size (and RAID level) recommendations depend on whether availability, performance, or cost is the priority for creating RAIDsets. Tradeoffs must be made because no single RAID level provides the perfect balance of availability, performance, and cost. You need to determine what your priorities are before creating your RAIDsets or stripesets.

For availability and performance, it is important to put each RAIDset member on a different port (bus). This keeps the RAIDset from going inoperative in the event of a single port failure, and also provides better performance.

RAID level 5 is more economical for large RAIDsets than smaller RAIDsets because the cost of the parity blocks is amortized across a larger number of devices. However, large RAIDsets statistically have higher failure rates.

Stripesets provide high performance and a lower cost (no parity disk to buy), but do not provide redundancy for availability.

Mirrorsets provide maximum availability and potentially improved read performance, but at the greatest cost.

7

Error Analysis and Fault Isolation

This chapter describes the errors, faults, and significant events that may occur during SWXRC-04 controller initialization and normal operation. A translation of the events and, in most cases, information on how to respond to a specific event, are also given.

CAUTION

Do not attempt to replace or repair components within FRUs or equipment damage may result. Use the controller fault indications and error logs to isolate FRU-level failures.

7.1 Special Considerations

Some or all of the situations presented in the following sections may apply when your controller detects a fault.

7.1.1 Nonredundant Configurations

When a controller (or its cache module, or both) fails in a nonredundant configuration, a short period of system down time is needed to remove the faulty unit and install a replacement. The devices attached to that controller are off line for the duration of the remove and replace cycle.

7.1.2 Dual-Redundant Configurations

When a controller fails in a dual-redundant configuration, there is no system down time because failover takes place; the surviving controller takes over service to the failed controller's ports and devices. Fault isolation and corrective actions are similar to a nonredundant configuration.

7.1.3 Cache Module Failures

If a cache module fails, its controller still functions using on-board cache; however, Digital recommends that you replace the cache module as soon as possible.

7.1.4 Write-Back Cache Battery Failures

When a write-back cache module’s batteries fail or are low, the controller alters its operations until the batteries are fully charged:

- Stripesets and disk-based units with write-back caching enabled are accessed in write-through (read cache) mode.
- 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 hagve access to good batteries is access in write-through (read cache) mode.
- If the battries go low after the controller initializes, 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.

7.2 Types of Error Reporting

The controller can notify you of an error through one or more of the following means:

- The OCP
- Device LEDs
- Environmental Monitor Unit (EMU)
- Error messages at a host virtual terminal, or error messages at a maintenance terminal (if attached)

7.3 Troubleshooting Basics

When an error occurs, use the following steps as top-level guidelines for fault isolation:

1. Make a note of all visual indicators (OCP, device LEDs, EMU, or error messages) available to you.
2. For surviving controllers in dual-redundant pairs, try entering the RESTART OTHER_CONTROLLER command. The surviving controller may be keeping its companion from operating.
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.

   See Sections 7.4 through 7.7 for detailed information about errors and repair actions.

> **CAUTION**
>
> Do not use a controller failure situation as an opportunity to move devices or otherwise reconfigure your subsystem. Doing so prevents the controller from communicating with its units once the fault is corrected.

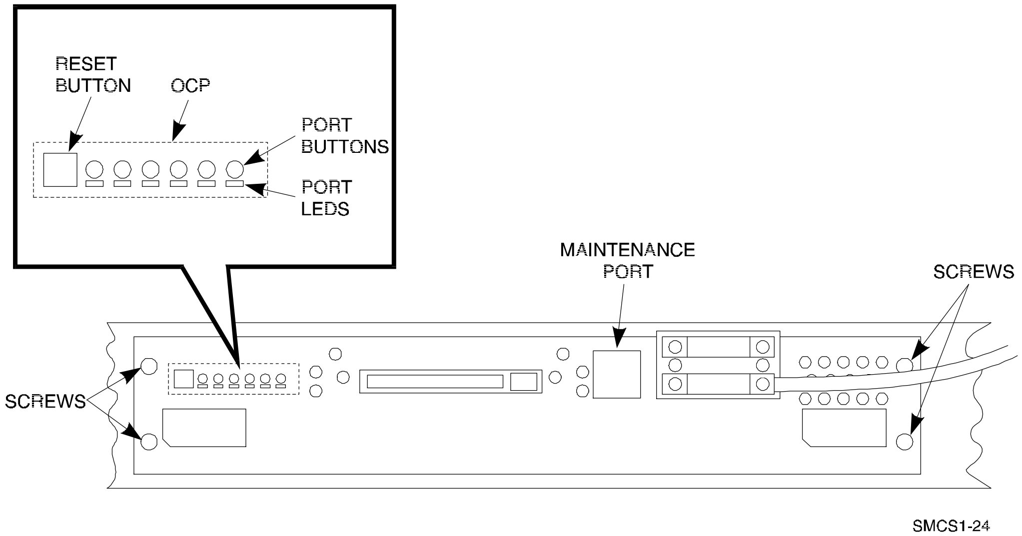
7.4 Operator Control Panel

The OCP includes the following:

- One reset button with an embedded green LED
- One button per SCSI port
- Six amber LEDs

Figure 7-1 shows the OCP from the SWXRC-04 controller. The buttons and LEDs serve different functions with respect to controlling the SCSI ports and/or reporting fault and normal conditions. Button and LED functions are discussed in the following sections.

Figure 7–1 Operator Control Panel



7.4.1 Normal Operation

The green LED (//) reflects the state of the controller and the host interface. Once controller initialization completes and its firmware is functioning, the green button flashes continuously at 1 Hz. Pressing the green button during this normal operation resets the controller.

Under normal operation, the amber LEDs indicate the state of the respective SCSI-2 buses attached to the controller.

When the devices on the buses are functioning correctly, the amber LEDs are not lit or flashing.

Pressing one of the port buttons at this time lights its corresponding amber LED and momentarily quiesces its SCSI-2 port. You must quiesce a port to remove or warm swap a device on the SCSI-2 bus for that port. See Chapter 9 for a detailed description of removing and replacing devices.

7.4.2 OCP Fault Notification

The OCP LEDs display information when the SWXRC-04 controller encounters a problem with a device configuration, a device, or the controller itself.

Should a configuration mismatch or a device fault occur, the amber LED for the affected device's bus lights continuously.

For controller problems, LED codes determined by internal diagnostics and operating firmware indicate either controller or operating firmware program card faults. In either case, the single (green) reset (//) LED lights continuously when an error is detected. The remaining (amber) LEDs display the error codes in two different ways:

- The error code is lit continuously for faults detected by internal diagnostic and initialization routines. See Figure 7-2 for the meaning of these codes, and the suggested maintenance actions that correspond to them.
- The error code flashes at 3 Hz representing faults that occur during normal controller operation. See Figure 7-3 for the meaning of these codes, and the suggested maintenance actions that correspond to them.

NOTE

Any flashing OCP codes present while initialization or self-test diagnostic error information is posted to the display momentarily stop flashing (becoming solidly lit). Normal flashing resumes once the display completes. Do not attempt to look up OCP codes while this information is printing at the terminal.

Figure 7–2 Solid OCP Codes

Reset	1	2	3	4	5	6		Description of Error	Action
●	●	●	●	●	●	●	3F	DAEMON hard Error.	Replace controller module.
●	●	●	●	●	●	○	3E	Repeated firmware bugcheck.	Replace controller module.
●	●	●	●	●	○	●	3D	NVMEM version mismatch.	Replace program card with later version.
●	●	●	●	●	○	○	3C	NVMEM write error.	Replace controller module.
●	●	●	●	○	●	●	3B	NVMEM read error.	Replace controller module.
●	●	●	●	○	●	○	3A	NMU error within firmware bugcheck.	RESET (//) the controller.
●	●	●	●	○	○	●	39	Inconsistent NVMEM structures repaired.[1]	RESET (//) the controller.
●	●	●	●	○	○	○	38	Bugcheck with no restart.	RESET (//) the controller.
●	●	●	○	●	●	●	37	Firmware induced to restart failed bugcheck failed to occur.	Replace controller module.
●	●	●	○	●	●	○	36	Hardware induced restart following bugcheck failed to occur.	Replace controller module.
●	●	●	○	●	○	●	35	Bugcheck within bugcheck controller.	RESET (//) the controller.
●	○	○	○	○	○	○	00	No program card seen.[2]	Replace controller module.

○=Off ●=Lit continuously

DAEMON=Diagnostic and Execution Monitor

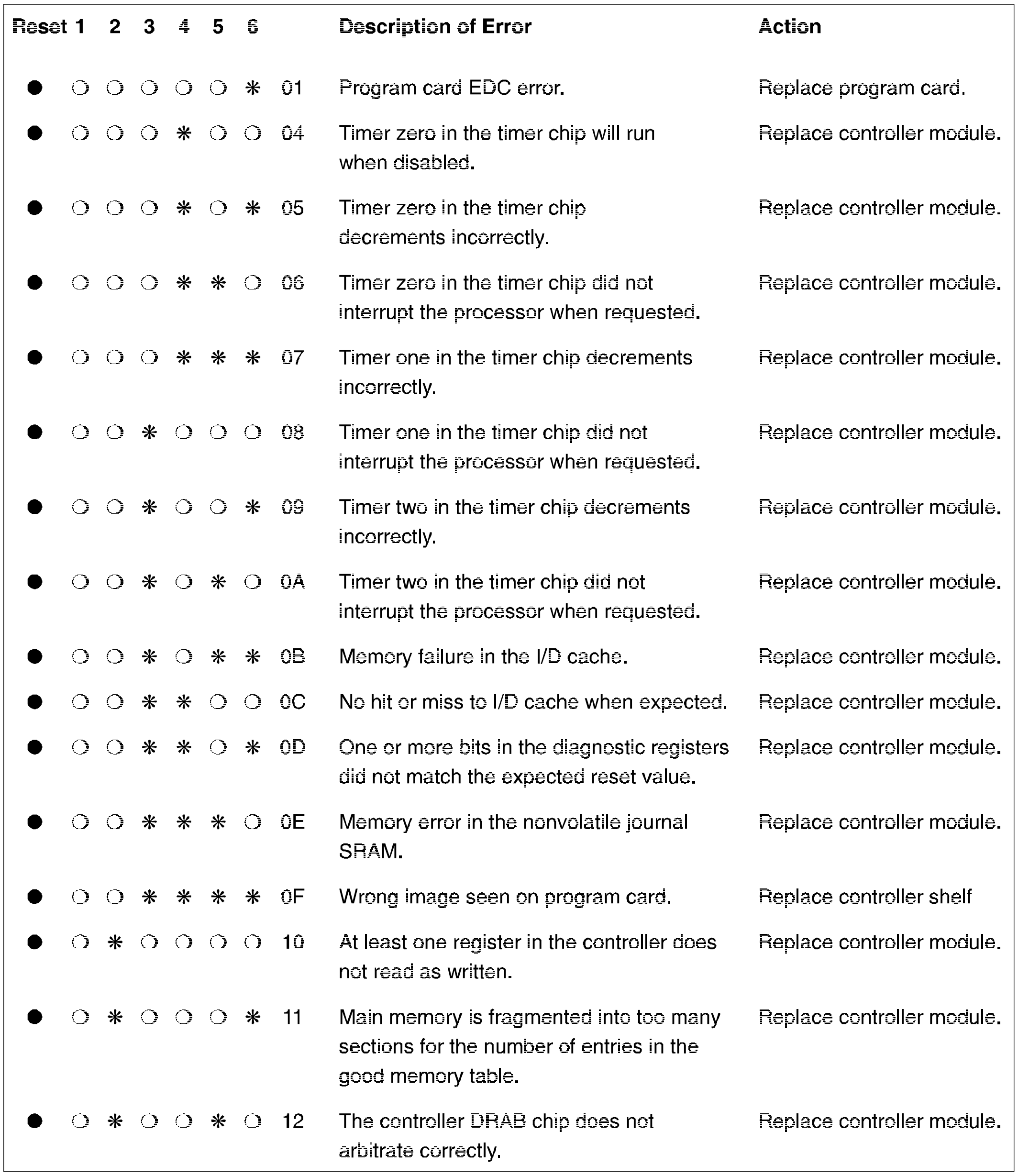
NVMEM=Nonvolatile Memory

NMI=Nonmaskable Interrupt

1. A power failure or controller reset during an NVMEM update caused this error. If the error occurs on one controller in a dual-redundant configuration, a configuration mismatch will probably occur upon restart.
2. Try the card in another module. If the problem moves with the card, replace the card. If the problem does not move with the card, replace the controller module.

SMCS1-14

Figure 7–3 Flashing OCP Codes



Reset	1	2	3	4	5	6		Description of Error	Action
●	○	○	○	○	○	✻	01	Program card EDC error.	Replace program card.
●	○	○	○	✻	○	○	04	Timer zero in the timer chip will run when disabled.	Replace controller module.
●	○	○	○	✻	○	✻	05	Timer zero in the timer chip decrements incorrectly.	Replace controller module.
●	○	○	○	✻	✻	○	06	Timer zero in the timer chip did not interrupt the processor when requested.	Replace controller module.
●	○	○	○	✻	✻	✻	07	Timer one in the timer chip decrements incorrectly.	Replace controller module.
●	○	○	✻	○	○	○	08	Timer one in the timer chip did not interrupt the processor when requested.	Replace controller module.
●	○	○	✻	○	○	✻	09	Timer two in the timer chip decrements incorrectly.	Replace controller module.
●	○	○	✻	○	✻	○	0A	Timer two in the timer chip did not interrupt the processor when requested.	Replace controller module.
●	○	○	✻	○	✻	✻	0B	Memory failure in the I/D cache.	Replace controller module.
●	○	○	✻	✻	○	○	0C	No hit or miss to I/D cache when expected.	Replace controller module.
●	○	○	✻	✻	○	✻	0D	One or more bits in the diagnostic registers did not match the expected reset value.	Replace controller module.
●	○	○	✻	✻	✻	○	0E	Memory error in the nonvolatile journal SRAM.	Replace controller module.
●	○	○	✻	✻	✻	✻	0F	Wrong image seen on program card.	Replace controller shelf
●	○	✻	○	○	○	○	10	At least one register in the controller does not read as written.	Replace controller module.
●	○	✻	○	○	○	✻	11	Main memory is fragmented into too many sections for the number of entries in the good memory table.	Replace controller module.
●	○	✻	○	○	✻	○	12	The controller DRAB chip does not arbitrate correctly.	Replace controller module.

○ = Off ● = Lit continuously ✻ = Flashing
I/D = Instruction/Data (cache on the controller module)
DRAB = Dynamic RAM Controller and Arbitration Engine (operates controller shared memory)
ECC = Error Correction Code
EDC = Error Detection Code
SRAM = Static RAM
NXM = Nonexistent Memory

SMCS1-11

Figure 7–3 Flashing OCP Codes (continued)

Reset	1	2	3	4	5	6		Description of Error	Action
●	○	✻	○	○	✻	✻	13	Controller DRAB chip failed to detect forced parity, or detected parity when not forced.	Replace controller module.
●	○	✻	○	✻	○	○	14	Controller DRAB chip failed to verify the EDC correctly.	Replace controller module.
●	○	✻	○	✻	○	✻	15	Controller DRAB chip failed to report forced ECC.	Replace controller module.
●	○	✻	○	✻	✻	○	16	Controller DRAB chip failed some operation in the reporting, validating, and testing of the multibit ECC memory error.	Replace controller module.
●	○	✻	○	✻	✻	✻	17	Controller DRAB chip failed some operation in the reporting, validating, and testing of the multiple single-bit ECC memory error.	Replace controller module.
●	○	✻	✻	○	○	○	18	Controller main memory did not write correctly in one or more sized memory transfers.	Replace controller module.
●	○	✻	✻	○	○	✻	19	Controller did not cause an I-to-N bus time-out when accessing a "reset" host port chip.	Replace controller module.
●	○	✻	✻	○	✻	○	1A	Controller DRAB chip did not report an I-to-N bus time-out when accessing a "reset" host port chip.	Replace controller module.
●	○	✻	✻	○	✻	○	1B	Controller DRAB chip did not interrupt the controller processor when expected.	Replace controller module.
●	○	✻	✻	✻	○	○	1C	Controller DRAB did not report an NXM error when nonexistent memory was accessed.	Replace controller module.
●	○	✻	✻	✻	○	✻	1D	Controller DRAB did not report an address parity error when one was forced.	Replace controller module.
●	○	✻	✻	✻	✻	○	1E	There was an unexpected nonmaskable interrupt from the controller DRAB during the DRAB memory test.	Replace controller module.
●	○	✻	✻	✻	✻	✻	1F	Diagnostic register indicates there is no cache module, but an interrupt exists from the nonexistent cache module.	Replace controller shelf backplane.

○ = Off ● = Lit continuously ✻ = Flashing
I/D = Instruction/Data (cache on the controller module)
DRAB = Dynamic RAM Controller and Arbitration Engine (operates controller shared memory)
ECC = Error Correction Code
EDC = Error Detection Code
SRAM = Static RAM
NXM = Nonexistent Memory

SMCS1-12

Figure 7–3 Flashing OCP Codes (continued)

Reset	1	2	3	4	5	6		Description of Error	Action
●	✻	❍	❍	❍	❍	❍	20	The required amount of memory available for the code image to be loaded for the program card is insufficient.	Replace controller module.
●	✻	❍	❍	❍	❍	✻	21	The required amount of memory available in pool area is insufficient for controller to run.	Replace controller module.
●	✻	❍	❍	❍	✻	✻	23	The required amount of memory available in buffer area is insufficient for controller to run.	Replace controller module.
●	✻	❍	❍	✻	❍	❍	24	The code image was not the same as the image on the card after the contents were copied to memory.	Replace controller module.
●	✻	❍	❍	✻	❍	✻	25	Diagnostic register indicates that the module exists, but access to that cache module caused an error.	Replace controller module.
●	✻	❍	❍	✻	✻	❍	26	Diagnostic register indicates that the module does not exist, but access to that cache module did not cause an error.	Replace controller module.
●	✻	✻	❍	❍	❍	❍	30	The journal SRAM is bad.	Replace controller module.
●	✻	✻	✻	❍	✻	❍	3A	There was an unexpected interrupt from a read cache or the present and lock bits are not working correctly.	Replace controller module.
●	✻	✻	✻	❍	✻	✻	3B	There is an interrupt pending to the controller's policy processor when there should be none.	Replace controller module.
●	✻	✻	✻	✻	❍	❍	3C	There was an unexpected fault during initialization.	Replace controller module.
●	✻	✻	✻	✻	❍	✻	3D	There was an unexpected maskable interrupt received during initialization.	Replace controller module.
●	✻	✻	✻	✻	✻	❍	3E	There was an unexpected nonmaskable interrupt received during initialization.	Replace controller module.
●	✻	✻	✻	✻	✻	✻	3F	An illegal process was activated during initialization.	Replace controller shelf backplane.

❍ = Off ● = Lit continuously ✻ = Flashing
I/D = Instruction/Data (cache on the controller module)
DRAB = Dynamic RAM Controller and Arbitration Engine (operates controller shared memory)
ECC = Error Correction Code
EDC = Error Detection Code
SRAM = Static RAM
NXM = Nonexistent Memory

SMCS1-13

7.5 Storage Building Block Indicators

Storage device and power supply SBBs have LEDs to indicate power and status. You can use these LEDs in conjunction with the OCP indicators to isolate certain faults, as discussed in the following sections.

7.5.1 Storage Device SBB Status

Each storage device SBB has two LED indicators that display the device's status. These LEDs have three states: on, off, and flashing.

- Device Activity LED – The Device Activity LED (green) is on or flashing when the device is active. The device SBB controls the Device Activity LED directly.

CAUTION

Do not remove a storage SBB when the Device Activity LED is on or flashing. This can cause the loss or corruption of data.

- Device Fault LED – The Device Fault LED (amber) indicates a device error condition when it is either on or flashing. The controller monitors the device's status and controls the fault LED using the StorageWorks high-availability storage subsystem fault bus.

  When the Device Fault LED indicates a fault, the amber, controller OCP LED for the device's port is lit continuously. You should record which devices have lit/flashing fault LEDs before resetting the controller, as a reset may temporarily clear the LED even though the fault remains.

Figure 7-4 shows the locations of the LEDs on the 3.5-inch SBBs. Table 7-1 shows the states of the device SBB LEDs, along with the device/shelf conditions they represent.

Figure 7–4 3.5-inch Storage SBB LEDs

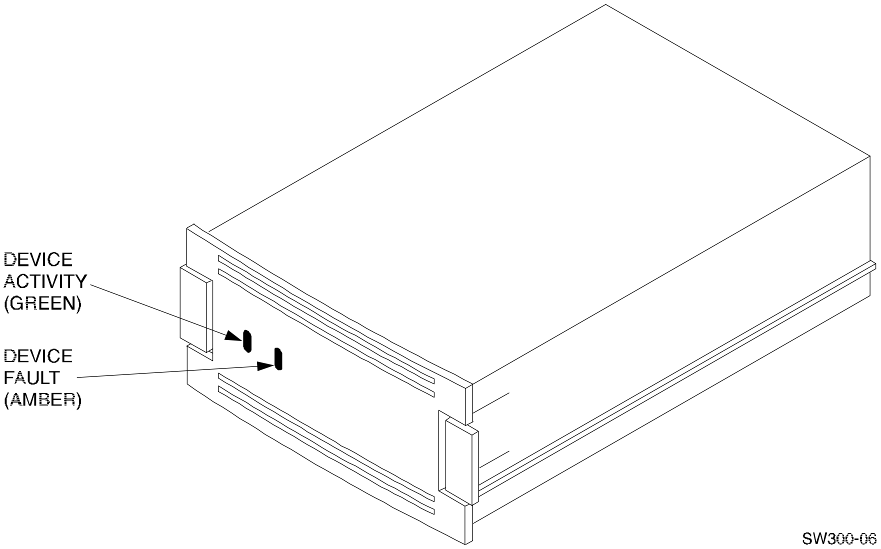


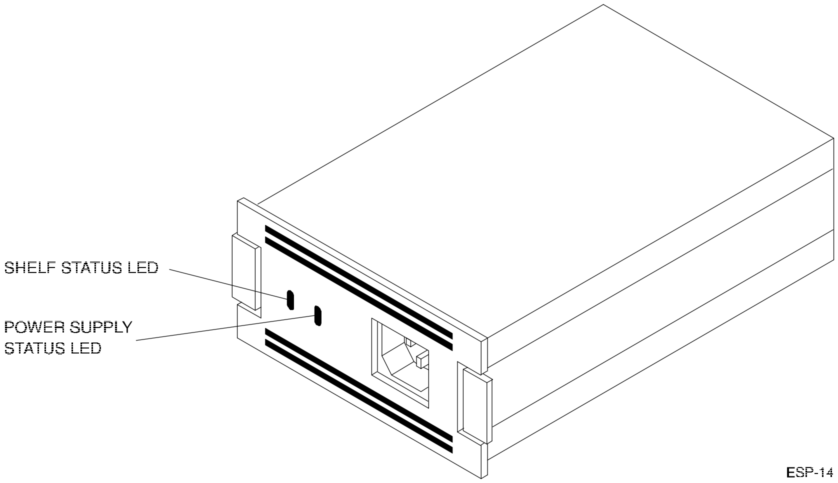
Table 7–1 Storage SBB Status LEDs

LED	Status	Indication
Device activity Device fault	On Off	SBB is operating normally.
Device activity Device fault	Flashing Off	SBB is operating normally.
Device activity Device fault	Off Off	SBB is operating normally. The SBB is inactive, and there is no fault.
Device activity Device fault	On On	Fault status. SBB is probably not responding to control signals. Digital recommends that you replace the SBB.
Device activity Device fault	Off On	Fault status. SBB is inactive and spun down. Digital recommends that you replace the SBB.
Device activity Device fault	On Flashing	Fault Status. SSB is active and is spinning down because of the fault.

7.5.2 Power Supply SBB and Shelf Status

The status of both the shelf blowers and power supplies is displayed on the power supply LEDs, as shown in Figure 7-5. The upper LED displays the shelf status and the lower LED displays the power supply SBB's status.

Figure 7–5 Power Supply LEDs



7.5.2.1 Single Power Supply LED Status

In a shelf with only one power supply, Table 7-2 shows the states of the power supply LEDs, along with their corresponding power supply/shelf status.

Table 7–2 Single Power Supply Status LEDs

Status LED	State	Indication
Shelf (upper) PS (lower)	On On	System is operating normally.
Shelf (upper) PS (lower)	Off On	Blower or power supply fault. There is a shelf fault; there is no power supply fault. Replace blower as described in Chapter 9.
Shelf (upper) PS (lower)	Off Off	Fault Status. Shelf or power supply fault. Replace power supply as described in Chapter 9.

7.5.2.2 Dual Power Supply LED Status

In a shelf with two power supplies, Table 7-3 shows the states of the LEDs on both power supplies, along with their corresponding power supply/shelf status.

Table 7–3 Dual Power Supply Status LEDs

Status LED	PS1 †	PS2 ‡	Indication
Shelf (upper) PS (lower)	On On	On On	Normal status. System is operating normally.
Shelf (upper) PS (lower)	Off On	Off On	Fault status. There is a shelf fault; there is no power supply fault. Replace blower as described in Chapter 9.
Shelf (upper) PS (lower)	Off On	Off Off	Fault Status. PS1 is operational. Replace PS2 as described in Chapter 9.
Shelf (upper) PS (lower)	Off Off	Off On	Fault status. PS2 is operational. Replace PS1 as described in Chapter 9.
Shelf (upper) PS (lower)	Off Off	Off Off	Fault Status. Possible PS1 and PS2 fault or input power problem.

† Shelf power supply installed in slot 7.
‡ Redundant power supply installed in slot 6.

7.6 Environmental Monitor Unit

The EMU is used only in the SWXSC-AA cabinet and provides increased protection against catastrophic subsystem faults. The EMU works with the controller to warn you of various existing or impending subsystem failures. The controller responds to such conditions by displaying console error messages, and by controlling warning LEDs on the EMU and the devices themselves.

The EMU performs the following specific functions (SWXSC-AA cabinets only):

- Monitors and controls the shelf blowers
- Monitors the on/off condition of each power supply
- Senses shelf/cabinet temperature
- Monitors power supply voltages

See the *StorageWorks SWXSC-AA Office Expansion RAID Enclosure User's Guide* for more information on the EMU.

7.6.1 EMU Fault Detection

Once the SWXSC-AA cabinet is powered on, it operates normally until a fault condition is detected by the EMU. If the EMU detects a fault condition, it performs any or all of the following:

- Turn on the appropriate LED on the EMU panel
- Turn on the fault (amber) LED on the front upper right corner of the SWXSC-AA cabinet
- Activate an audible alarm
- Increase the speed of the blowers

As shown in Figure 7-6, the EMU front panel LEDs display the information when the subsystem is turned on or encounters a problem. Table 7-4 lists the EMU control panel buttons and LEDs, their functions, and error descriptions.

Figure 7–6 EMU Control Panel LEDs and Switches

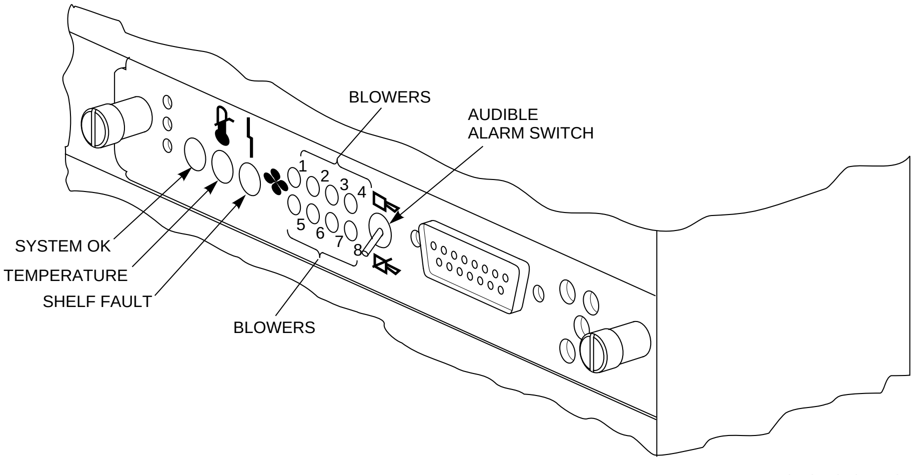


Table 7–4 EMU Control Panel LEDs/Switches and Error Descriptions

When the ...	Is	The subsystem ...
System OK (green) LED Shelf Fault (amber) LED Blower 1 through 8 (amber) LEDs Temperature (amber) LED Audible Alarm	ON OFF OFF OFF OFF	Is in the normal operating state.
Audible Alarm	ON	Has a failed FRU. Check the EMU control panel LEDs to determine which FRU has failed.
System OK (green) LED	OFF	Has a failed FRU and one or more of the following error conditions exist: The temperature is above the maximum safe operating level. The +12 V dc power supply output is out of regulation. The +5 V dc power supply output is out of regulation. Fewer than four power supply SBBs are operational. One controller may be defective. Check for any error messages on the terminal (if one is attached).
Blower *x LED*	ON	Has an individual blower *x* failure.
Blower 1, 2, 3, *and* 4 (amber) LEDs	ON	Has one of these four blowers either not installed or not connected.
Blower 5, 6, 7, *and* 8 (amber) LEDs	ON	Has one of these four blowers either not installed or not connected.
Temperature (amber) LED	ON	Operating temperature is too high. The blowers operate at high speed to reduce the temperature.
Shelf Fault (amber) LED	ON	Has one or more of the following faults: A power supply SBB has failed. (Verify if any individual power supply SBB LEDs are off to confirm.) A storage device SBB or one of the controllers has failed. Refer to Section 7.4 for additional information on status LEDs on the controller OCP.
		Fewer than four power supply SBBs are operational.

7.6.2 Controller Fault Detection

If the controller detects a storage device failure or a fault in the other controller in a dual-redundant configuration, it uses a controller fault drive signal to illuminate the fault (amber) LED on the EMU. When a special "controller alive" signal from either controller indicates to the EMU that a controller is not functioning, the EMU:

- turns on its fault (amber) LED
- turns off its System OK (green) LED
- activates an audible alarm

The SWXRC-04 controller generates an error message on the maintenance terminal (see Section 7.8.3) when it detects one of the following fault conditions:

- a power supply SBB has failed
- a blower has failed or is either not installed or not connected
- The operating temperature is too high

7.7 Error and Status Messages

The SWXRC-04 controller outputs error and status messages in response to both system- and operator-initiated events. Spontaneous error and status messages occur at any time, and are driven by system events. Interactive errors are associated with some action that the operator is performing.

7.8 Spontaneous Error and Status Messages

Spontaneous error and status messages are part of a CLI error report (CER) system, which causes the error message text to appear on a maintenance terminal along with the CLI prompt, as shown in the following example:

```
%CER -- 13-JUL-1994 13:28:45 -- SWAP signal cleared - all
SWAP interrupts reenabled

CLI>

%CER -- 13-JUL-1994 13:29:11 -- Other controller restarted

CLI>
```

Spontaneous CER messages appear only under the following conditions:

- A hardware maintenance terminal is connected for receiving messages. Spontaneous messages do not appear on virtual terminals.
- The subsystem has finished initializing.
- You are not currently running a utility program on the maintenance terminal.
- A 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 a maintenance terminal.

If any one of the previous conditions is not met, the spontaneous CER message display does not occur. In this case, the SWXRC-04 operating firmware stores messages for you. You need only connect a virtual or maintenance terminal (if one is not already connected) and press the Return key from the CLI prompt to review the 15 most recently received error messages.

Often, 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 they cannot be recalled afterwards.)

> **NOTE**
>
> Because the severity of errors varies, the controller may not initialize or operate, or both, even though an error message appears.
>
> For example, if all of the SCSI ports, or the host port and local terminal port fail diagnostics, the controller cannot operate. However, if the cache module fails during normal operation, the controller continues to operate.

The following sections list spontaneous error and status messages you may encounter. The CER messages are presented in the following error categories:

- Configuration and CLI
- Diagnostic and initialization
- EMU
- Failover
- NVPM
- Read cache
- Shelf
- Write-back cache

Consult your firmware release notes for updates to the list of error messages.

7.8.1 Spontaneous Configuration and CLI Messages

This section lists messages that appear when configuration inconsistencies occur.

```
Configuration information deleted due to internal
inconsistencies
```

Explanation: This message displays if a test of nonvolatile memory shows corruption. The configuration information for the controller is deleted when this message is displayed.

```
Controllers misconfigured. Type SHOW THIS_CONTROLLER
```

Explanation: If this message appears, examine the SHOW THIS_CONTROLLER display to determine the source of the misconfiguration.

```
Device and/or storageset names changed to avoid conflicts
```

Explanation: Digital adds new CLI keywords at each new SWXRC-04 operating firmware release that can conflict with existing device and/or storageset names. When this happens, SWXRC-04 operating 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.

```
Licensing different between the two controllers
```

Explanation: The licensing features are set differently on two controllers of a dual-redundant pair.

```
Restart of the other controller required
```

Explanation: When changing some parameters, you must reinitialize the companion controller in a dual-redundant pair to have the parameter take effect.

```
Restart of this controller required
```

Explanation: A changed parameter requires reinitialization of this controller to take effect.

```
Serial number initialized due to format error
```

Explanation: An invalid serial number was entered for the second controller of a dual-redundant pair.

```
Taken out of failover due to serial number format error
```

Explanation: An invalid serial number format was entered for the second controller of a dual-redundant pair.

```
Restart of the controller required to apply new patch
```

Explanation: You used the Code Patch utility to enter a firmware patch, but the patch is not applied until you restart the controller.

7.8.2 Spontaneous Diagnostic and Initialization Messages

This section contains error messages that may be displayed if a fault occurs during initialization or self-test diagnostics. See Chapter 8 for more information on diagnostics.

```
Half CACHE FAILED Diagnostics
```

Explanation: Up to 50% of the cache memory has failed diagnostic tests.

```
HOST port FAILED Diagnostics
```

Explanation: The host port of the controller has failed diagnostics.

```
Local Terminal Port FAILED Diagnostics
```

Explanation: The maintenance (EIA-423) terminal port has failed diagnostics.

```
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.

```
Whole CACHE FAILED Diagnostics
```

Explanation: The cache module has failed diagnostics tests.

```
Write-Back CACHE BATTERY FAILED Diagnostics, cache treated
as READ CACHE
```

Explanation: The write-back cache battery has failed the diagnostic tests described in Chapter 8. Any unwritten cache data on the cache is flushed. Non-RAIDset units are accessed in read cache mode. RAIDsets are unavailable unless they have access to good batteries on a dual-redundant companion cache. (An exception is if CACHE_POLICY=B is used. For more information about CACHE_POLICY=B, refer to the command, SET THIS_CONTROLLER CACHE_POLICY=B.) Replace the cache battery.

7.8.3 Spontaneous EMU Messages

This section contains error messages that may be displayed if an EMU-detected fault occurs (SWXSC-AA shelves only). Refer to Section 7.6 for more information on the EMU.

`Power Supply failure detected`

Explanation: One of the power supply SBBs in the SWXSC-AA shelf has failed.

`Fan failure detected`

Explanation: A blower in the SWXSC-AA shelf has failed.

`WARNING: High temperature detected`

Explanation: The SWXSC-AA shelf temperature is above the recommended operating temperature of 35°C (95°F). The blowers run at high speed. If the shelf temperature rises over 50°C (122°F), the subsystem shuts down.

`Power Supply failure cleared`

Explanation: A power supply SBB fault is corrected.

`Fan failure cleared`

Explanation: A blower fault is corrected.

`Temperature within optimum limit`

Explanation: The SWXSC-AA shelf temperature has returned to below 35°C (95°F). If the blowers were on high speed, they now return to normal speed.

7.8.4 Spontaneous Failover Messages

The messages in this section are generated during failover between dual-redundant controllers.

`Both HSxxx controllers are using SCSI address 6`

Explanation: There is a hardware problem with the SWXSS-01 shelf. This problem probably involves the shelf backplane.

`Both HSxxx controllers are using SCSI address 7`

Explanation: There is a hardware problem with the SWXSS-01 shelf. This problem probably involves the shelf backplane.

`Other controller not responding - RESET signal asserted`

Explanation: One controller in a dual-redundant configuration is locked up, not responding, or the kill line to it is asserted.

`Other controller restarted`

Explanation: The other controller in a dual-redundant pair has successfully restarted after failing or undergoing a bugcheck.

`Received LAST GASP message from other controller`

Explanation: One controller in a dual-redundant configuration is attempting an automatic restart after failing or undergoing a bugcheck.

`SCSI Device and HSxxx 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.

7.8.5 Spontaneous NVPM Messages

The messages listed in this section are displayed because of a problem or fault associated with the nonvolatile parameter memory (NVPM).

NOTE

Some NVPM messages read "NVPM *component-name* component initialized to default settings." For some of these initialization cases, corrective action clears the error message only until the next time the controller is reset, because the error could be caused by a fault in NVPM itself. If the error persists, replace the controller module.

`NVPM Failover Information component initialized to default settings.`

Explanation: The identity of the other controller in a dual-redundant pair has been lost. Enter the SET FAILOVER COPY=OTHER_CONTROLLER command to correct this problem.

NVPM Revision level updated from *n* to *N*.

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.

`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 information recorded at installation time, and use the SET THIS_CONTROLLER and SET OTHER_CONTROLLER commands to restore terminal settings.

```
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
```

2. Compare the information displayed with the configuration information recorded at installation time or with a copy of the most current configuration.
3. Reconfigure the necessary devices, units, or storagesets. (See the CLI commands described in Appendix B.)

```
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.

```
NVPM FMU Parameters component initialized to default
settings.
```

Explanation: FMU settings have been lost. Default FMU options are in place until you run FMU to change them.

```
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, "SWXRC-04") the controller module should be replaced immediately.

CAUTION

Replace the controller immediately if any of the following NVPM messages occur. Do not continue to use the controller.

```
All NVPM components initialized to their default
settings.
```

```
Controller Characteristics component reformat failed
during NVPM Revision Level 1 to 2 reformat.
```

```
Host Access Disabled.
```

```
NVPM Controller Characteristics component initialized to
default settings.
```

```
NVPM Recursive Bugcheck Information component
initialized to default settings.
```

```
NVPM System Information Page component initialized to
default settings.
```

```
NVPM Volume Serial Number component initialized to
default settings.
```

```
The following NVPM Manufacturing Failure Information
component elements were initialized to default settings:
[...list of component elements
```

```
Unknown NVPM Revision Level.
```

```
Unknown reformat stage encountered during NVPM Revision
Level 1 to 2 reformat.
```

7.8.6 Spontaneous Read Cache Messages

This section contains error messages that may be displayed if a read cache related fault occurs.

```
Cache module failed diagnostic testing
```

Explanation: The cache has failed the diagnostic tests described in Chapter 8. Replace the read cache module.

```
Cache module failed diagnostic testing - half not
accessible
```

Explanation: Up to 50% of the cache memory has failed the diagnostic tests described in Chapter 8. Replace the read cache module.

7.8.7 Spontaneous Shelf Messages

This section lists messages that appear when a shelf problem occurs.

```
Shelf xx fixed
```

Explanation: Shelf number *xx* has been correctly repaired.

```
Shelf xx has a bad power supply or fan
```

Explanation: Troubleshoot the system to isolate and replace the failed component.

```
SWAP signal cleared - all SWAP interrupts re-enabled
```

Explanation: This message indicates that the swap signal is now cleared.

```
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.

7.8.8 Spontaneous Write-Back Cache Messages

This section contains error messages that may be displayed if a write-back cache related fault occurs.

```
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 can also 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 enables you to use the random module.

```
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 that 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 enables you to use the cache module (although you lose the unwritten cache data).

```
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 enables you to run (in write-through mode) without a cache but without accessing RAIDsets.

```
Cache module failed diagnostic testing
```

Explanation: The cache has failed the diagnostic tests described in Chapter 8. Any unwritten cache data in the cache module is lost. Replace the write-back cache module.

```
Cache module failed diagnostic testing - half not
accessible
```

Explanation: Up to 50 percent of the cache memory has failed the diagnostic tests described in Chapter 8. Unwritten cache data in the cache module is lost. Replace the write-back cache module.

```
Cache battery failed diagnostic testing
```

Explanation: The write-back cache battery has failed the diagnostic tests described in Chapter 8. Any unwritten cache data on the cache is flushed. Non-RAIDset units are accessed in read cache mode. RAIDsets are unavailable unless they have access to good batteries on a dual-redundant companion cache. Replace the cache battery.

```
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 are accessed in read cache mode. RAIDsets are unavailable unless they have access to good batteries on a dual-redundant companion cache. (An exception is if CACHE_POLICY=B is used. For more information about CACHE_POLICY=B, refer to the command, SET THIS_CONTROLLER CACHE_POLICY=B.) Replace the cache battery.

```
Cache modules are misconfigured
```

Explanation: This message is generated in dual-redundant configurations under the following circumstances:

- The companion controller's cache module is not a write-back cache.
- The companion controller's write-back cache is not the same size as this controller's write-back cache.

Cache failover of unwritten cache data is not performed if this message occurs. Correct the problem by replacing/adding cache to make sure both are compatible.

7.8.9 Last Failure Logging Messages

If you used the Fault Management Utility (FMU) to enable spontaneous last failure logging (LFL) displays, you may see maintenance terminal messages that begin with "%LFL." With LFL display enabled, the controller spontaneously displays information relevant to the sudden termination of executing firmware.

In cases when 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.

See Chapter 8 for more information on the FMU and last failure logging.

7.8.10 Event Logging Messages

If you used FMU to enable spontaneous event logging (EVL) display, you may see maintenance terminal messages that begin with "%EVL." With EVL display enabled, the controller spontaneously displays EIP information during your maintenance terminal session.

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 do not appear when the terminal again becomes available. (The %EVL display is lost.)

See Chapter 8 for more information on the FMU and event logging.

7.9 Interactive Error and Status Messages

The previous sections detailed spontaneous, system-initiated error and status messages you may encounter. For a list of messages you may see during while interacting with the controller via the CLI, see Appendix B.

8

Diagnostics, Exercisers, and Utilities

This chapter describes firmware programs available to assist in the operation and diagnosis of the SWXRC-04 controller subsystem.

This chapter discusses the following firmware programs:

- Initialization and self-test routines
- Disk exerciser (DILX)
- System performance utility (VTDPY)
- Configuration utility (CONFIG)
- Menu-driven Configuration utility (CFMENU)
- Code Load/Code Patch utility (CLCP)
- Firmware Licensing System (FLS)
- Fault Management Utility (FMU)
- Crash Utility
- Volume Serial Number Utility (CHVSN)
- CLONE

8.1 Initialization

The SWXRC-04 controller executes a number of diagnostic programs as part of its initialization sequences. The controller initializes after any of the following conditions:

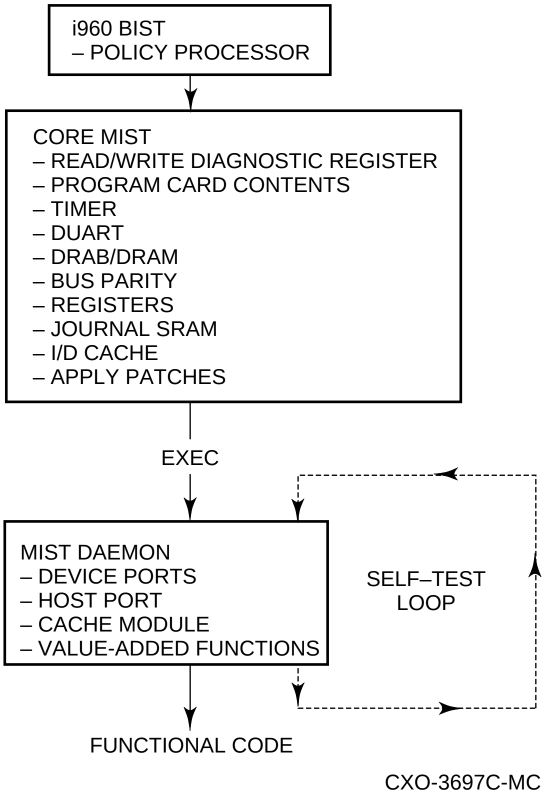
- Power is turned on.
- The firmware resets the controller.
- The operator presses the green reset (//) button.
- The controller receives a RESTART command

Whenever the controller initializes, it steps through a three-phase series of tests designed to detect any hardware or firmware faults. The three test areas are as follow:

1. Built-in self-test
2. Core module integrity self-test
3. Module integrity self-test DAEMON

Initialization time varies. However, initialization always completes in under 1 minute. Figure 8-1 shows the initialization process.

Figure 8–1 Controller Initialization



8.1.1 Built-In Self-Test

The controller begins initialization by executing its policy processor's internal built-in self-test (**BIST**). BIST always executes upon initialization, because it is an integral part of the i960 microcode. BIST runs entirely from the i960 chip and a small portion of the firmware program card.

Successful completion of BIST means the i960 chip is functioning properly. If BIST fails, the controller shows no activity, and all port indicators on the OCP are off. (The green reset LED is solidly lit.) BIST fails if an incorrect program card is present.

8.1.2 Core Module Integrity Self-Test

After BIST completes successfully, initialization routines and diagnostics expand to testing of the controller module itself. The tests are part of the program card firmware and are known as core module integrity self-test (MIST).

Just before beginning core MIST, the controller reads the initial boot record (IBR) to determine the address of hardware setup parameters and process control information. After reading the IBR, the firmware within the program card is initialized to the IBR parameters. Program card firmware then executes core MIST as follows:

1. MIST checks the initial state of the read/write diagnostic register.
2. The test validates program card contents by reading each memory location and computing an error detection code (EDC). The test then compares the computed EDC with a predetermined EDC. The program card contents are valid if both EDCs match.
3. Core MIST then tests and/or checks module hardware attached to the buses:
    - Timer operation
    - DUART operation
    - DRAB/DRAM (shared memory) operation

      The test writes to and reads all legal addresses. Then, boundaries are checked by attempting to access nonexistent addresses. To pass this test, the first two megabytes of memory must test good. If bad segments are found, the bad segments may divide total memory into no more than 16 good, continuous sections.

      The test selects a device, then checks whether or not the bus has selected that device.

      The test verifies that each allowable memory transfer size works and that illegal transfer sizes do not.
    - Bus parity
    - Registers (The test checks registers for frozen bits.)
    - Journal SRAM (The test writes to and reads all journal SRAM addresses.)
    - I/D cache
4. After core MIST successfully tests the program card and bus hardware, the initialization routine loads the firmware into the first two megabytes of controller shared memory. The initialization routine then uses the EDC method to compare the memory contents with the program card to verify a successful download.
5. The initialization routine applies any memory-resident code patches for this firmware version, and updates the free memory list accordingly.
6. The i960 is initialized to the new parameters (the ones read from the IBR). At this time, control of initialization passes to the firmware executive (EXEC). EXEC runs from controller shared memory.

If, at any time, a fault occurs during core MIST, the OCP displays a code (refer to Chapter 7). In addition, register contents are displayed at the terminal. The register contents should be reported to your service provider to help with module troubleshooting and repair.

The following is an example of MIST diagnostic error output:

```
MIST Diag-err Error Code  0000000F
MIST Diag-err Parameter 1 01234567
MIST Diag-err Parameter 2 89ABCDEF
MIST Diag-err Parameter 3 00000001

MIST Diag-err Master/Cache DRAB register contents:
MIST Diag-err   DSR  2D170CBC, 2C17003D
MIST Diag-err   CSR  00000000, 00000000
MIST Diag-err   DCSR 00000000, 00003402
MIST Diag-err   DER  00001C00, 00001C76
MIST Diag-err   EAR  00800000, 00000000
MIST Diag-err   EDR  00800000, FFFFFFFF
MIST Diag-err   ERR  0000002F, 00000000
MIST Diag-err   RSR  09805432, 09805432
MIST Diag-err   CHC  00000000, 00000000
MIST Diag-err   CMC  00078006, 00000000

MIST Diag-err Read diagnostic register 0  E7FFFFFC
MIST Diag-err Read diagnostic register 1  FFFFFEF0
MIST Diag-err Write diagnostic register 0 C3FFF80F
MIST Diag-err Write diagnostic register 1 0002FF04

MIST Diag-err Halting.
```

8.1.3 Module Integrity Self-Test DAEMON

Once initialization control is passed to EXEC, EXEC calls the diagnostic and execution monitor (DAEMON). DAEMON tests the device port hardware, host port hardware, cache module, and value-added functions.

- To test the device ports, DAEMON checks each NCR 53C710 SCSI processor chip. Initialization continues unless all SCSI device ports fail testing. In other words, it is possible for the controller to run with only one functioning device port.
- DAEMON tests the host port hardware for the controller. The NCR 53C720 host processor chip is tested. Initialization continues even if the host port tests fail, however, DAEMON stops initialization if the dual universal asynchronous receiver transmitter (DUART) test (from core MIST) and the host port tests fail.
- DAEMON tests the cache module as follows:

NOTE

The controller still functions if the cache module fails its testing. In this case, the controller uses its on-board shared memory for caching operations.

- **Read/Write-back cache**

  DAEMON works in close conjunction with the cache manager, a program that runs in the background of controller firmware. DAEMON first asks the cache manager if unwritten cache data exists on the write-back cache.

  If unwritten cache data does not exist, DAEMON tests the DRAB (memory controller) on the cache module, followed by testing the write-back cache batteries. After DAEMON completes, and functional code takes control of the firmware, the cache manager tests the memory on the cache. At least the first megabyte of the memory must test good, or the cache is declared bad.

  If unwritten cache data exists, DAEMON tests only the batteries, and postpones other tests. Later, in operational code, the cache manager flushes the unwritten cache data after testing the DRAB and memory.

  If cache is locked by the other controller (dual-redundant configurations), then all cache DAEMON diagnostics are postponed. During functional code, when the cache manager determines that the cache is unlocked, the cache manager tests the DRAB, batteries, and memory.

  The tests run by DAEMON and the cache manager are summarized in Table 8-1.

Table 8-1 Cache Module Testing

Test	DAEMON	Cache Manager
DRAB	• All memory is initialized • Full address test	• No memory is initialized. • Address test on diagnostic pages only
Memory	• Never invoked	• Always invokes all memory tests. • Read only, or read/write.
Battery	• If battery is charged, full battery test.	• If battery is charged, full battery test except non-DRAB DRAM refresh circuitry is not tested.
	• Otherwise, battery interrupt is checked.	• Otherwise, battery interrupt is checked.

  DAEMON handles all interrupts and errors received during cache module testing. If DAEMON receives any interrupt, it stops initialization. DAEMON and/or the cache manager display any errors as a code on the OCP (refer to *Chapter 7*).

- To make sure of proper RAID functions, DAEMON tests the XOR operations of the value-added hardware and the associated buffer memory. Initialization stops if this test fails.

After successful test completion, DAEMON releases control. At this time, initialization is finished, and functional controller firmware takes over.

Self-Test Self-test is a special function of DAEMON, where you set DAEMON to run in a continuous loop. Self-test enables you to diagnose intermittent hardware failures because the loop continues until an error is detected. The DAEMON battery tests, however, only run once per self-test. They do not repeat or loop.

In addition, self-test checks the controller hardware without affecting devices on any ports. DIGITAL recommends you run self-test from the maintenance terminal because the host port disconnects once the controller begins self-test.

To run self-test, enter one of the following commands (which command you need depends on your configuration, which controller the terminal is connected to, and which controller you wish to test.)

```
CLI> SELFTEST THIS_CONTROLLER
CLI> SELFTEST OTHER_CONTROLLER
```

See Appendix B for more information on the command and its qualifiers.

When you run self-test, all outstanding I/O operations complete. The controller also attempts to flush the cache. However, even if self-test fails to flush the cache, the program continues to execute.

Self-test halts if it detects a fault. Otherwise, the self-test loop continues until you press the reset (//) button or cycle the controller power off and on, after which the controller reinitializes.

8.2 Disk Inline Exerciser

DILX is a diagnostic tool used to exercise the data transfer capabilities of selected disks connected to an SWXRC-04 controller. DILX exercises disks in a way that simulates a high level of user activity. Using DILX, you can read from all customer-available data areas and write to all single-device units. Thus, DILX can be used to determine the health of a controller and the disks connected to it and to acquire performance statistics. You can run DILX from a maintenance terminal. DILX allows for auto-configuring of drives. This allows for quick configuring and testing of all units at once.

WARNING

Customer data will be lost by running this test. DIGITAL recommends only using the Auto-Configure option during initial installations.

DILX performs read tests on logical units that may consist of storagesets of multiple physical devices. Error reports identify the logical units, not the physical devices. Therefore, if errors occur while running against a unit, its storageset must be reconfigured as individual devices, and then DILX run again, against the individual devices. When run against individual devices, DILX performs both read and write tests.

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.

8.2.1 Invoking DILX

To invoke DILX from a maintenance terminal, enter the following command at the CLI> prompt:

```
CLI>    RUN DILX
```

8.2.2 Interrupting DILX Execution

Use the following guidelines to interrupt DILX execution.

> **NOTE**
>
> The symbol "^" is used to indicate the Ctrl key in this guide and in the controller's firmware messages. 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.

8.2.3 DILX Tests

The two DILX tests are the Basic Function Test and the User-Defined Test.

8.2.3.1 Basic Function Test — DILX

The Basic Function test for DILX executes in two or three phases. The three phases are as follow:

- **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 re-executed no matter how long the DILX execution time is set. The other phases are re-executed on a 10-minute cycle.
- **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 reads and writes in the ratio specified previously under read/write ratio. When read-only mode is chosen, only read commands are issued.

If compares are enabled, compares 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 re-executed 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-->
```

- **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 re-executed at 10-minute intervals with each cycle approximately 2 minutes.

8.2.3.2 User-Defined Test — DILX

CAUTION

The User-Defined test should 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 data 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.

8.2.4 DILX 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:

```
It is recommended that DILX only be run when there is no
host activity present on the SWXRC-04 controller. Do you
want to continue (y/n) [n] ?
```

The following message describing the Auto-Configure option is displayed:

```
The Auto-Configure option automatically selects, for
testing, all of the disk units configured. It performs a
very thorough test  with *WRITES* enabled. The user is
only able to select the run time and performance summary
options. The user is not able to specify specific units
to test. The Auto-Configure option is only recommended
for initial installations. It is the first question
asked.
```

```
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 displays the following caution statement:

```
**CAUTION**
All data on the Auto-Configured disks is destroyed. You
*MUST* be sure of yourself.
```

```
Are you sure you want to continue (y/n) [n] ?
```

Explanation: This question is self explanatory.

```
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 = 10 minutes.
- Performance summary interval = 10 minutes.
- Displaying sense data for hard or soft errors is disabled.
- The hard error limit = 65535. Testing stops if the limit is reached.
- The I/O queue depth = 4. A maximum of 4 I/Os is outstanding at any time.

- The Selected Test = the Basic Function test.
- Read-only mode.
- All user available LBNs are available for testing.
- Data compares are disabled.

```
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.

```
Enter the 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.

```
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.

```
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.

```
When the hard error limit is reached, the unit is 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.

```
When the soft error limit is reached, soft errors are no
   longer displayed but testing continues 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.

```
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.

```
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 D107 would be specified as unit number 107.

```
Unit x are 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.

```
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 displays the following test selections:

```
***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 then displays the following messages:

```
In the User-Defined test, you may define up to 20
commands.  They are executed in the order entered.  The
commands are repeated until the execution time limit
expires.

** CAUTION **
If you define write commands, user data is 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 selection(s), the following message is displayed by DILX:

```
* 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 is destroyed. *Be sure of your actions before answering this question.* This question applies to all DILX tests. Enter "N" to enable read only mode, where read and access commands are the only commands enabled.

```
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 writes 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.

```
The write percentage is 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 enables 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 system.

```
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 writes 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 8.2.8 for a list of data patterns.) The default uses all 18 patterns in a random method. This question also enables you to create a unique data pattern of your own choice.

```
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.

```
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.

```
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.

```
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 reads. This option can be used to test data integrity.

```
Enter compare percentage (1:100) [5] ?
```

Explanation: This question is displayed only if you choose to perform data compares. This question enables you to change the percentage of read and write commands that have a data compare operation performed. Enter a value indicating the compare percentage. The default is 5.

```
Enter command number x (read, write, quit) []  ?
```

Explanation: This question only applies to the User-Defined test. It enables you to define command x as a read, write, access, or erase command. Enter quit to finish defining the test.

```
Enter starting LBN for this command
(0:highest_lbn_on_the_disk) [] ?
```

Explanation: This question only applies to the User-Defined test. It enables you to set the starting LBN for the command currently being defined. Enter the starting LBN for this command.

```
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 enables 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.

```
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 performs 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 is 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.

```
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.

```
The new unit is 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.

8.2.5 DILX Output Messages

The following message is displayed when DILX is started:

```
Disk Inline Exerciser - version 2.0
```

This message identifies the internal program as DILX and gives the DILX software version number.

```
Change Unit is not a legal option if Auto-Configure was
chosen.
```

Explanation: This message is displayed if the user selected the Auto-Configure option and selected 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.

```
DILX – Normal Termination.
```

Explanation: This message is displayed when DILX terminates under normal conditions.

```
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:

- `Unable to allocate memory.`

  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.

- `Cannot perform tests.`

  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.

- `Unable to change operation mode to maintenance.`

  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, submit a CLD (error report), then reset the controller.

```
Disk unit x does not exist.
```

Explanation: An attempt was made to allocate a unit for testing that does not exist on the controller.

`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.

`Unable to allocate unit.`

Explanation: This message should be preceded by a reason why the unit could not be allocated for DILX testing.

`DILX detected error, code x.`

Explanation: The "normal" way DILX recognizes an error on a unit is through the reception of SCSI sense data. However, the following are some errors that DILX detects using internal checks without SCSI sense data:

- `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 were 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 have 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 failed and what the data were 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 a Ctrl/Y (termination request), a Ctrl/G (print summary request) or a 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.

`DILX does 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.)

`Unable to bring unit online.`

Explanation: This message is self explanatory.

```
Soft error reporting disabled. Unit x.
```

Explanation: This message indicates that the soft error limit has been reached and therefore no more soft errors are displayed for this unit.

```
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.

```
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.

```
Hard error limit reached for controller errors. All units
dropped from testing.
```

Explanation: This message is self explanatory.

```
Unit is already allocated for testing.
```

Explanation: This message is self explanatory.

```
No drives selected.
```

Explanation: DILX parameter collection was exited without choosing any units to test.

```
Maximum number of units are now configured.
```

Explanation: This message is self explanatory. (Testing starts after this message is displayed.)

```
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.

```
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.

```
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.

```
Disk unit numbers on this controller include:
```

Explanation: After this message is displayed, a list of disk unit numbers on the controller is displayed.

```
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.

```
DILX terminated prematurely by user request.
```

Explanation: A Ctrl/Y was entered. DILX interprets this as a request to terminate. This message is displayed and DILX terminates.

```
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.

```
This unit is reserved.
```

Explanation: The unit could not be allocated for testing because a host has reserved the unit.

```
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.

```
The unit does not have any media present.
```

Explanation: The unit could not be allocated for testing because no media is present.

```
The RUNSTOP_SWITCH is set to RUN_DISABLED.
```

Explanation: The unit could not be allocated for testing because the RUNSTOP_SWITCH is set to RUN_DISABLED. This is enabled and disabled through the Command Line Interpreter (CLI).

```
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:

```
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
```

8.2.6 DILX Sense Data Display

To interpret the sense data fields correctly, refer to SCSI-2 specifications. Example 8-1 is an example of a DILX sense data display.

Example 8–1 DILX Sense Data Display

```
Sense data in hex for unit x

   Sense Key     x
   Sense ASC     x
   Sense ASQ     x
   Instance      x
```

8.2.7 DILX Deferred Error Display

Example 8-2 is an example of a DILX deferred error display.

Example 8–2 DILX Deferred Error Display

```
Deferred error detected, hard error counted against each
unit.

   Sense Key     x
   Sense ASC     x
   Sense ASQ     x
   Instance      x
```

8.2.8 DILX Data Patterns

Table 8-2 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 8–2 DILX Data Patterns

Pattern Number	Pattern in hex
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

Table 8–2 DILX Data Patterns (continued)

Pattern Number	Pattern in hex
11	2D2D, 2D2D, 2D2D, D2D2, D2D2, D2D2, 2D2D, 2D2D, D2D2, D2D2, 2D2D, 2D2D, 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
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	

8.2.9 Interpreting the DILX 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 or not performance statistics are requested in the user-specified parameters and if errors are detected.

The following is an example of a DILX performance display where performance statistics were not selected and where 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
```

The following is an example of a DILX performance display where performance statistics were selected and 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
```

The following is an example of a DILX performance display where performance statistics were not selected and where errors were detected on a unit under test:

```
                DILX Summary at 18-JUN-1993 06:18:41
                Test minutes remaining: 0, expired: 6

❶ Unit 10      Total IO Requests 153259
            No errors detected

❷ Unit 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 ErrsHard Cnt 0     Soft Cnt 1

❸  Unit 55     Total IO Requests 201719
            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

❹           Total Errs   Hard Cnt 0   Soft Cnt 2
```

where:

❶ Represents the unit number and the total I/O requests to this unit.

❷ Represents the unit number and total I/O requests to this unit.

This also includes the following items associated with this error, and the total number of hard and soft errors for this unit:

- The SWXRC-04 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

❸ Represents information about the first two unique errors for this unit.

This also includes the following items associated with this error, and the total number of hard and soft errors for this unit:

- The SWXRC-04 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.

❹ Represents the total hard and soft errors experienced for this unit.

The following is an example of a DILX performance display where performance statistics were not selected and where 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.

- IC–The SWXRC-04 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.

8.2.10 DILX Abort Codes

Table 8-3 lists the DILX abort codes and definitions.

Table 8–3 DILX Abort Codes and Definitions

Value	Definition
1	An IO 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 in	The semaphore was set after a oneshot I/O was issued but nothing was found 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 ^Y.

8.2.11 DILX Error Codes

Table 8-4 list the DILX error codes and definitions for DILX-detected errors.

Table 8–4 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.

8.3 VTDPY Utility

The VTDPY utility gathers and displays system state and performance information for the SWXRC-04 family of modular storage controllers. 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 also can be used. VTDPY can be run only on terminals connected to the SWXRC-04 controller maintenance terminal port.

The following sections show how to use the VTDPY utility.

8.3.1 How to Run VTDPY

Only one VTDPY session can be run on each controller at one time. Prior to running VTDPY, be sure the terminal is set in NOWRAP mode. Otherwise, the top line of the display scrolls off of the screen.

To initiate VTDPY from a maintenance terminal at the CLI> prompt, enter the following command:

```
CLI> RUN VTDPY
```

8.3.2 Using the VTDPY Control Keys

Use the following control key sequences to work the VTDPY display:

Table 8–5 VTDPY Control Keys

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, without updating information. (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 with current information (same as Ctrl/G).

8.3.3 Using the VTDPY Command Line

VTDPY contains a command line interpreter that is invoked by entering Ctrl/C any time after the program has begun execution. 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 8.3.2.

Table 8–6 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, without updating the information.
QUIT	Terminates program (same as EXIT).
UPDATE	Updates screen display with current information.

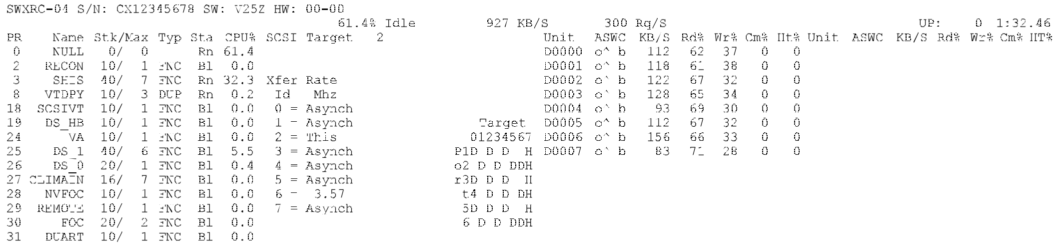
The keywords in the command strings can be abbreviated to the minimum number of characters that are necessary to uniquely identify the keyword. Typing a question mark (?) after a keyword causes the parser to provide a list of keywords or values that can follow the supplied keyword. The command line interpreter is not case sensitive.

Upon successful execution of a command other than HELP, the command line interpreter is exited and the display is resumed. Typing 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.

8.3.4 How to Interpret the VTDPY Display Fields

This section describes the major fields in the VTDPY displays. Examples of the VTDPY screens are shown followed by an explanation of each field of the screens.

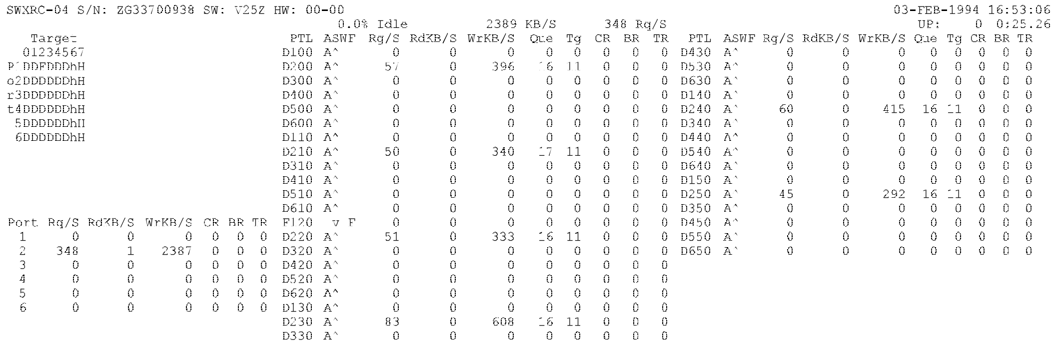
Figure 8–2 VTDPY Default Display



```
SWXRC-04 S/N: CX12345678 SW: V25Z HW: 00-00
                                          61.4% Idle         927 KB/S       300 Rq/S                                UP:    0  1:32.46
PR    Name Stk/Max Typ Sta CPU% SCSI Target   2                      Unit  ASWC  KB/S Rd% Wr% Cm% Ht% Unit  ASWC KB/S Rd% Wr% Cm% HT%
 0    NULL   0/  0      Rn 61.4                                      D0000 o^ b   112  62  37   0   0
 2   RECON  10/  1 FNC  Bl  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   Mhz                            D0003 o^ b   128  65  34   0   0
18  SCSIVT  10/  1 FNC  Bl  0.0  0 = Asynch                          D0004 o^ b    93  69  30   0   0
19   DS_HB  10/  1 FNC  Bl  0.0  1 = Asynch                   Target D0005 o^ b   112  67  32   0   0
24      VA  10/  1 FNC  Bl  0.0  2 = This                   01234567 D0006 o^ b   156  66  33   0   0
25    DS_1  40/  6 FNC  Bl  5.5  3 = Asynch               P1D D D  H D0007 o^ b    83  71  28   0   0
26    DS_0  20/  1 FNC  Bl  0.4  4 = Asynch               o2 D D DDH
27 CLIMAIN  16/  7 FNC  Bl  0.0  5 = Asynch               r3D D D  H
28   NVFOC  10/  1 FNC  Bl  0.0  6 =  3.57                 t4 D D DH
29  REMOTE  10/  1 FNC  Bl  0.0  7 = Asynch                5D D D  H
30     FOC  20/  2 FNC  Bl  0.0                            6 D D DDH
31   DUART  10/  1 FNC  Bl  0.0
```

SMCS1-20

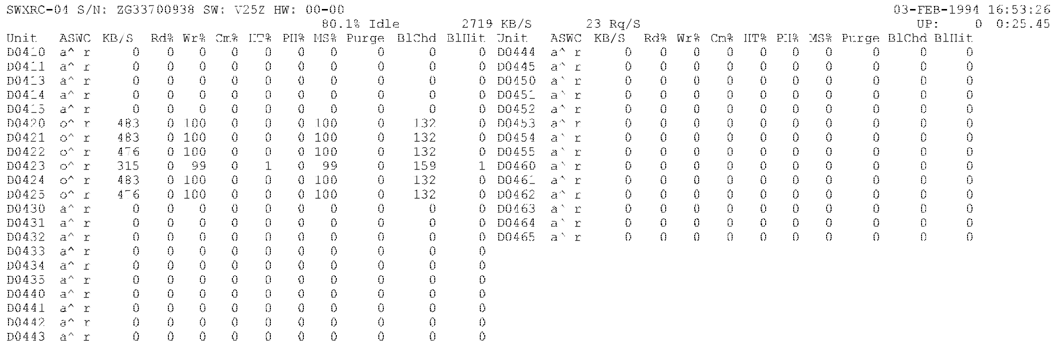
Figure 8–3 VTDPY Device Performance Display



```
SWXRC-04 S/N: ZG33700938 SW: V25Z HW: 00-00                                                                      03-FEB-1994 16:53:06
                                          0.0% Idle          2389 KB/S      348 Rq/S                                UP:    0  0:25.26
   Target                           PTL ASWF  Rq/S RdKB/S  WrKB/S  Que Tg  CR  BR  TR  PTL ASWF Rq/S  RdKB/S WrKB/S Que Tg CR BR TR
  01234567                         D100 A^       0      0       0    0  0   0   0   0 D430 A^      0       0      0   0  0  0  0  0
P1DDFDDDhH                         D200 A^      57      0     396   16 11   0   0   0 D530 A^      0       0      0   0  0  0  0  0
o2DDDDDDhH                         D300 A^       0      0       0    0  0   0   0   0 D630 A^      0       0      0   0  0  0  0  0
r3DDDDDDhH                         D400 A^       0      0       0    0  0   0   0   0 D140 A^      0       0      0   0  0  0  0  0
t4DDDDDDhH                         D500 A^       0      0       0    0  0   0   0   0 D240 A^     60       0    415  16 11  0  0  0
 5DDDDDDhH                         D600 A^       0      0       0    0  0   0   0   0 D340 A^      0       0      0   0  0  0  0  0
 6DDDDDDhH                         D110 A^       0      0       0    0  0   0   0   0 D440 A^      0       0      0   0  0  0  0  0
                                   D210 A^      50      0     340   17 11   0   0   0 D540 A^      0       0      0   0  0  0  0  0
                                   D310 A^       0      0       0    0  0   0   0   0 D640 A^      0       0      0   0  0  0  0  0
                                   D410 A^       0      0       0    0  0   0   0   0 D150 A^      0       0      0   0  0  0  0  0
                                   D510 A^       0      0       0    0  0   0   0   0 D250 A^     45       0    292  16 11  0  0  0
                                   D610 A^       0      0       0    0  0   0   0   0 D350 A^      0       0      0   0  0  0  0  0
Port Rq/S RdKB/S  WrKB/S CR BR TR  F120   v F    0      0       0    0  0   0   0   0 D450 A^      0       0      0   0  0  0  0  0
  1     0      0       0  0  0  0  D220 A^      51      0     333   16 11   0   0   0 D550 A^      0       0      0   0  0  0  0  0
  2   348      1    2387  0  0  0  D320 A^       0      0       0    0  0   0   0   0 D650 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
```

SMCS1-21

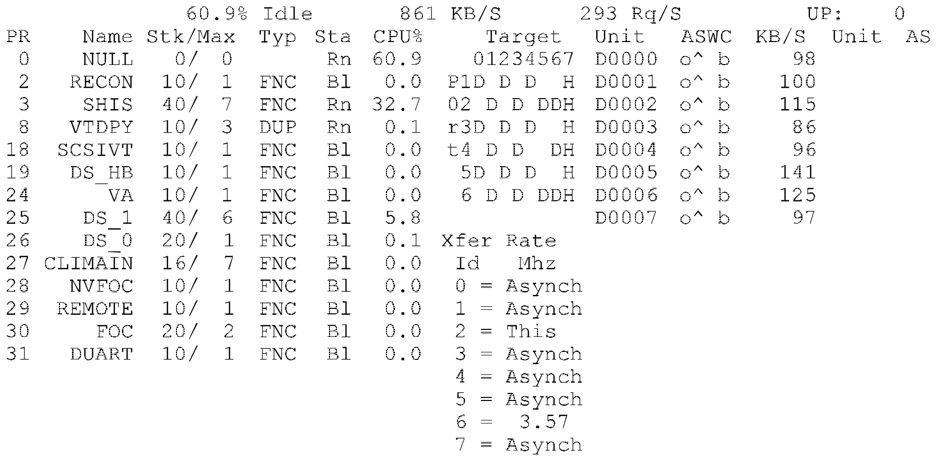
Figure 8–4 VTDPY Unit Cache Performance Display



```
SWXRC-04 S/N: ZG33700938 SW: V25Z HW: 00-00                                                                      03-FEB-1994 16:53:26
                                        80.1% Idle        2719 KB/S       23 Rq/S                                   UP:    0  0:25.45
Unit   ASWC KB/S  Rd% Wr% Cm% HT% PH% MS% Purge BlChd  BlHit Unit   ASWC KB/S  Rd% Wr% Cm% HT% PH% MS% Purge BlChd  BlHit
D0410  a^ r     0   0   0   0   0   0   0     0     0      0 D0444  a^ r     0   0   0   0   0   0   0     0     0      0
D0411  a^ r     0   0   0   0   0   0   0     0     0      0 D0445  a^ r     0   0   0   0   0   0   0     0     0      0
D0413  a^ r     0   0   0   0   0   0   0     0     0      0 D0450  a^ r     0   0   0   0   0   0   0     0     0      0
D0414  a^ r     0   0   0   0   0   0   0     0     0      0 D0451  a^ r     0   0   0   0   0   0   0     0     0      0
D0415  a^ r     0   0   0   0   0   0   0     0     0      0 D0452  a^ r     0   0   0   0   0   0   0     0     0      0
D0420  o^ r   483   0 100   0   0   0 100     0   132      0 D0453  a^ r     0   0   0   0   0   0   0     0     0      0
D0421  o^ r   483   0 100   0   0   0 100     0   132      0 D0454  a^ r     0   0   0   0   0   0   0     0     0      0
D0422  o^ r   476   0 100   0   0   0 100     0   132      0 D0455  a^ r     0   0   0   0   0   0   0     0     0      0
D0423  o^ r   315   0  99   0   1   0  99     0   159      1 D0460  a^ r     0   0   0   0   0   0   0     0     0      0
D0424  o^ r   483   0 100   0   0   0 100     0   132      0 D0461  a^ r     0   0   0   0   0   0   0     0     0      0
D0425  o^ r   476   0 100   0   0   0 100     0   132      0 D0462  a^ r     0   0   0   0   0   0   0     0     0      0
D0430  a^ r     0   0   0   0   0   0   0     0     0      0 D0463  a^ r     0   0   0   0   0   0   0     0     0      0
D0431  a^ r     0   0   0   0   0   0   0     0     0      0 D0464  a^ r     0   0   0   0   0   0   0     0     0      0
D0432  a^ r     0   0   0   0   0   0   0     0     0      0 D0465  a^ r     0   0   0   0   0   0   0     0     0      0
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
```

SMCS1-22

Figure 8–5 VTDPY Brief SCSI Status Display



```
              60.9% Idle       861 KB/S      293 Rq/S          UP:    0
PR    Name Stk/Max  Typ Sta  CPU%     Target   Unit  ASWC  KB/S  Unit  AS
 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  02 D D DDH  D0002 o^ b   115
 8   VTDPY  10/  3  DUP  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
```

SMCS1-23

Display Header

`SWXRC-04 ❶ S/N: CX00000002 ❷ SW: V25Z ❸ HW: A-02 ❹`

Description

This subdisplay provides title information for the display. For 132 column displays, this subdisplay is displayed across one line.

❶ Controller model

❷ Controller serial number

❸ Controller firmware version

❹ Controller hardware version

Date and Time

`29-JAN-1994 13:46:34 ❶`

`Up:    1  3:45.19 ❷`

Description

This subdisplay provides time information.

❶ System date and time. Only displayed if the operator has set these parameters in the controller.

❷ Time in days, hours, minutes, and seconds since the last controller boot.

Controller Performance Summary

```
47.2% Idle ❶ 1225 KB/S ❷  106 Rq/S ❸
```

Description

This subdisplay provides total system performance information.

❶ Policy processor idle rate.

❷ 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.

❸ 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 displayed, this is the request rate between the controller and the devices.

Controller Threads Display

```
Pr    Name    Stk/Max     Typ    Sta   CPU%
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
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
28    CLIMAIN 16/ 6       FNC    Bl    0.0
30    FOC     16/ 4       FNC    Bl    0.0
31    DUART   10/ 2       FNC    Bl    0.0
```

Description

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 are displayed.

- The **Pr** column lists the thread priority. The higher the number, the higher the priority.
- The **Name** column contains the thread name. For DUP Local Program threads, this is the name used to invoke the program.
- The **Stk** column lists the allocated stack size in 512 byte pages. The Max column lists the number of stack pages actually used.

- 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.
- 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.
- The **CPU%** column lists the percentage of execution time credited to each thread since the last screen update. The values may not add up to exactly 100 percent due to both rounding errors and the 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 8-7 describes the processes that may appear in the active thread display.

NOTE

It is possible that different versions of the controller firmware may have different threads or different names for the threads.

Table 8–7 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 SWXRC-04 array 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.
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.
REMOTE	The thread that manages state changes initiated by the other controller in a dual controller configuration.
RMGR	The thread that manages the data buffer pool.
RECON	The thread that rebuilds the parity blocks on RAID 5 storagesets when needed.
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.
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.

SCSI Host Port Characteristics

```
Xfer Rate
T❶W❷I❸Mhz❹
1 W 7 10.00
2 W Async❺
```

Description

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.

❶ SCSI host port target ID.

❷ Transfer width. W indicates 16 bit or wide transfers are being used. A space indicates 8 bit transfers are being used.

❸ The initiator with which synchronous communication has been negotiated.

❹ 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.

❺ **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.

Device SCSI Status

```
 Target
 01234567 ❶
P1 DDDDFhH ❷
o2TTT T hH
r3DDD   hH
t4DDDDDDhH
 5DDDD  hH
 6      hH❸
```

Description

This display shows what devices the controller has been able to identify on the device buses.

NOTE

The controller does not look for devices that are not configured into the nonvolatile memory using the CLI ADD command.

❶ 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.

❷ 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.

❸ 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.

Unit Status (abbreviated)

```
Unit    ASWC    KB/S     Rd%     Wr%     Cm%  HT%
D0110   a^ r    0        0       0       0    0
D0120   a^ r    0        0       0       0    0
D0130   o^ r    236      100     0       0    100
```

Description

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.

- 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.
- The ASWC columns 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 service provider. 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 disks, this symbol indicates the device is spinning up.
  - <–For disks, this symbol indicates the device is spinning down.
  - **v**–For disks, this symbol indicates the device is stopped.
  - 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.

- **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 units.
- **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 device types.
- **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 device types.
- **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 device types.
- **HT%**–This column indicates the cache hit percentage for data transferred between the host and the unit.

Unit Status (full)

```
Unit   ASWC  KB/S   Rd%  Wr%   Cm%   HT%  PH%  MS%   Purge  BlChd  BlHit
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
```

Description

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.

- 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.
- 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 disks, this symbol indicates the device is spinning up.
  - **<**–For disks, this symbol indicates the device is spinning down.
  - **v**–For disks, this symbol indicates the device is stopped.

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.

- **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.
- **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 device types.
- **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 device types.
- **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 device types.
- **HT%**–This column indicates the cache hit percentage for data transferred between the host and the unit.
- **PH%**–This column indicates the partial cache hit percentage for data transferred between the host and the unit.
- **MS%**–This column indicates the cache miss percentage for data transferred between the host and the unit.
- **Purge**–This column shows the number of blocks purged from the write back cache in the last update interval.
- **BlChd**–This column shows the number of blocks added to the cache in the last update interval.
- **BlHit**–This column shows the number of cached data blocks "hit" in the last update interval.

Device Status

```
 PTL  ASWF  Rq/S   RdKB/S   WrKB/S  Que   Tg  CR  BR  TR
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
D300   A^    11     93        0      2    1   0   0   0
D310   A^    0       0        0      0    0   0   0   0
D320   A^    36     247       0     12    10  0   0   0
D400   A^    11     93        0      2    1   0   0   0
D410   A^    0       0        0      0    0   0   0   0
D420   A^    36     247       0     10    8   0   0   0
D430   A^    0       0        0      0    0   0   0   0
D440   A^    0       0        0      0    0   0   0   0
D450   A^    0       0        0      0    0   0   0   0
D500   A^    11     93        0      1    1   0   0   0
D510   A^    0       0        0      0    0   0   0   0
D520   A^    0       0        0      0    0   0   0   0
D530   A^    47      0       375     6    5   0   0   0
```

Description

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.

- 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.
- 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 disks, this symbol indicates the device is spinning up.
- <–For disks, this symbol indicates the device is spinning down.
- **v**–For disks, this symbol indicates the device is stopped.
- For other types of devices, this column is left blank.

For disks, 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.

- **Rq/S**–This column shows the average I/O request rate for the device during the last update interval. These requests are up to 8 kilobytes long and are either generated by host requests or cache flush activity.
- **RdKB/S**–This column shows the average data transfer rate from the device in kilobytes during the previous screen update interval.
- **WrKB/S**–This column shows the average data transfer rate to the device in kilobytes during the previous screen update interval.
- **Que**–This column shows the maximum number of transfer requests waiting to be transferred to the device during the last screen update interval.
- **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.
- **CR**–This column indicates the number of SCSI command resets that occurred since VTDPY was started.
- **BR**–This column indicates the number of SCSI bus resets that occurred since VTDPY was started.
- **TR**–This column indicates the number of SCSI target resets that occurred since VTDPY was started.

Device SCSI Port Performance

```
Port   Rq/S   RdKB/S  WrKB/S  CR     BR     TR
 1      0       0       0      0      0      0
 2      11      93      0      0      0      0
 3      48      41      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
```

Description

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.

The Port column indicates the number of the SCSI device port.

- **Rq/S-**This column shows the average I/O request rate for the port during the last update interval. These requests are up to 8 kilobytes long and are either generated by host requests or cache flush activity.
- **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.
- **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.
- **CR-**This column indicates the number of SCSI command resets that occurred since VTDPY was started.
- **BR-**This column indicates the number of SCSI bus resets that occurred since VTDPY was started.
- **TR-**This column indicates the number of SCSI target resets that occurred since VTDPY was started.

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
performance display
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>
```

Description

This is the sample output from executing the HELP command.

8.4 CONFIG Utility

The CONFIG utility locates and adds devices to the controller. You should run the CONFIG utility whenever new devices are added to the controller.

8.4.1 CONFIG Conventions

The CONFIG utility searches all PTL device combinations to determine what devices exist on the subsystem. It adds all new devices that are found. The CONFIG utility does not initialize these devices, and it does not add units or storagesets.

8.4.2 Running the CONFIG Utility

You can run the CONFIG utility on either a virtual terminal or on a maintenance terminal. Before running the CONFIG utility, you may use the SHOW DEVICES command to verify the list of devices that are currently configured on the controller, as shown in the following example.

```
CLI> SHOW DEVICES
No devices
CLI> SHOW DEVICES

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 DISK120   1 2 0
add disk DISK140   1 4 0
add disk DISK210   2 1 0
add disk DISK230   2 3 0
add disk DISK500   5 0 0
add disk DISK520   5 2 0

Config - Normal Termination
CLI>

CLI> SHOW DEVICES

Name       Type      Port  Targ  LUN           Used by
----------------------------------------------------------

DISK100    disk        1     0    0
DISK120    disk        1     2    0
DISK140    disk        1     4    0
DISK210    disk        2     1    0
DISK230    disk        2     3    0
DISK500    disk        5     0    0
DISK520    disk        5     2    0
CLI>
```

After you run the CONFIG utility, you may have to initialize your containers using the INITIALIZE command as described in Appendix B.

8.5 Configuration Menu Utility

The configuration menu (CFMENU) enables you to quickly configure storage devices attached to the controller. CFMENU presents configuration commands normally entered at the CLI in a menu-driven format.

The CFMENU 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 also can be used. CFMENU can be run only on terminals connected the controller maintenance terminal port. Prior to running CFMENU, be sure the terminal is set in NOWRAP mode. Otherwise, the display may not correctly appear on the screen.

You should have a good understanding of the various CLI commands described in Appendix B before running CFMENU, because CFMENU asks you to choose options for devices, storagesets, and units based on the command qualifiers of the CLI.

This section presents an overview of one configuration, from start to finish. Only one example is described because of the many combinations of choices that can be made during any configuring situation.

8.5.1 Restrictions

The following restrictions apply to CFMENU:

- For dual-redundant configurations, do not run CFMENU on both controllers at the same time.
- There is currently no support for modifying switches (options) on existing devices, storagesets, or units. You must delete and re-add to modify options.
- Use CFMENU only for configuring storagesets. Specifically, do not use CFMENU to set WRITEBACK_CACHE or NOWRITEBACK_CACHE. Set these parameters through the CLI instead. Enabling caching through CFMENU may result in cache policies inconsistent with what CFMENU displays.

8.5.2 Main Menu

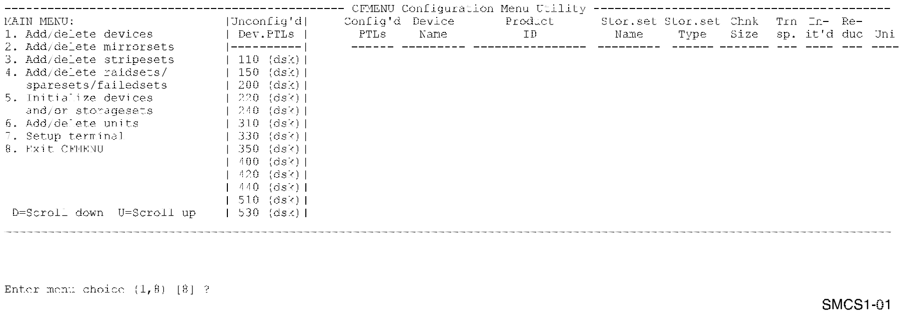
Run CFMENU during a terminal session by invoking it from the CLI prompt:

```
CLI> RUN CFMENU
```

The main menu appears as shown in Figure 8-6. Any attached devices that have not been added to your configuration appear, with their PTLs, in the column to the right of the menu options.

When prompted with "y/n/q", enter y for Yes, n for No, and q for Quit.

Figure 8–6 CFMENU Main Menu



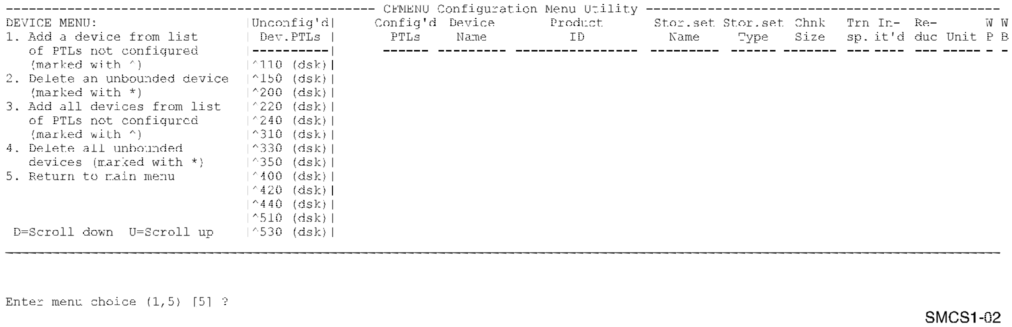
```
---------------------------------------------- CFMENU Configuration Menu Utility ----------------------------------------
MAIN MENU:                       |Unconfig'd|    Config'd  Device          Product        Stor.set Stor.set Chnk   Trn  In-  Re-
1. Add/delete devices            | Dev.PTLs |      PTLs     Name             ID             Name     Type    Size   sp. it'd duc  Uni
2. Add/delete mirrorsets         |----------|     ------ --------- ----------------  ---------  ------ ------- ---  ---- ---  ----
3. Add/delete stripesets         | 110 (dsk)|
4. Add/delete raidsets/          | 150 (dsk)|
   sparesets/failedsets          | 200 (dsk)|
5. Initialize devices            | 220 (dsk)|
   and/or storagesets            | 240 (dsk)|
6. Add/delete units              | 310 (dsk)|
7. Setup terminal                | 330 (dsk)|
8. Exit CFMENU                   | 350 (dsk)|
                                 | 400 (dsk)|
                                 | 420 (dsk)|
                                 | 440 (dsk)|
                                 | 510 (dsk)|
 D=Scroll down  U=Scroll up      | 530 (dsk)|
--------------------------------------------------------------------------------------------------------------------------


Enter menu choice (1,8) [8] ?
```

8.5.3 Adding Devices

From the main menu, enter option 1 (press 1 <Return>). The device menu appears as shown in Figure 8-7. The same list of unconfigured devices remains to the right of the options. When the list is too long to be shown on one screen, you can enter D or U to scroll the information down/up.

Figure 8–7 CFMENU Device Menu (Before Adding Devices)



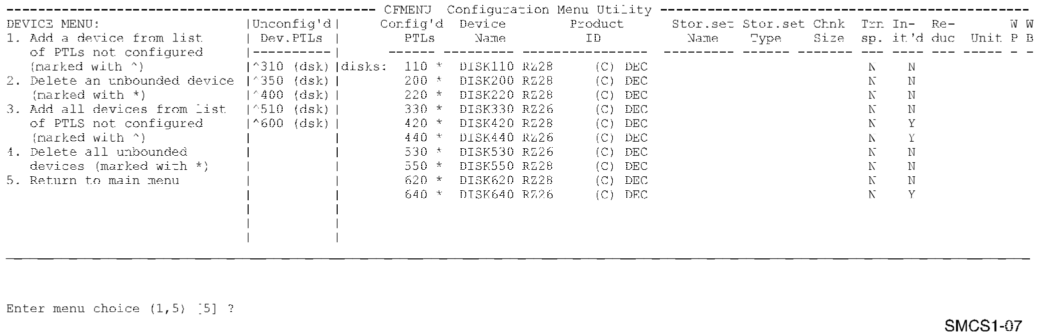
```
------------------------------------------------ CFMENU Configuration Menu Utility ----------------------------------------------
DEVICE MENU:                     |Unconfig'd|    Config'd  Device         Product        Stor.set Stor.set Chnk    Trn In-   Re-        W W
1. Add a device from list        | Dev.PTLs |      PTLs     Name            ID             Name     Type    Size    sp. it'd  duc  Unit P B
   of PTLs not configured        |----------|     ------ --------- ----------------  ---------  ------ -------  --- ----  --- ----- - -
   (marked with ^)               |^110 (dsk)|
2. Delete an unbounded device    |^150 (dsk)|
   (marked with *)               |^200 (dsk)|
3. Add all devices from list     |^220 (dsk)|
   of PTLs not configured        |^240 (dsk)|
   (marked with ^)               |^310 (dsk)|
4. Delete all unbounded          |^330 (dsk)|
   devices (marked with *)       |^350 (dsk)|
5. Return to main menu           |^400 (dsk)|
                                 |^420 (dsk)|
                                 |^440 (dsk)|
                                 |^510 (dsk)|
 D=Scroll down  U=Scroll up      |^530 (dsk)|
---------------------------------------------------------------------------------------------------------------------------------


Enter menu choice (1,5) [5] ?
```

In Figure 8-7, none of the devices have been added to the configuration yet. If you enter option 1, CFMENU asks you whether or not to add each device on the list. If you enter option 3, CFMENU adds all the unknown devices, in the same way as the CONFIG utility (refer to Section 8.4).

After entering option 1 and adding some devices, the screen resembles Figure 8-8.

Figure 8–8 CFMENU Device Menu



```
---------------------------------------------- CFMENU  Configuration Menu Utility ----------------------------------------------
DEVICE MENU:                  |Unconfig'd|     Config'd  Device        Product       Stor.set Stor.set Chnk  Trn In-  Re-      W W
1. Add a device from list     | Dev.PTLs |       PTLs     Name           ID            Name    Type    Size sp. it'd duc  Unit P B
   of PTLs not configured     |----------|      ------ --------- ----------------  --------- ------ ------- --- ---- --- ----- - -
   (marked with ^)            |^310 (dsk)|disks:  110 *  DISK110 RZ28     (C) DEC                          N   N
2. Delete an unbounded device |^350 (dsk)|        200 *  DISK200 RZ28     (C) DEC                          N   N
   (marked with *)            |^400 (dsk)|        220 *  DISK220 RZ28     (C) DEC                          N   N
3. Add all devices from list  |^510 (dsk)|        330 *  DISK330 RZ26     (C) DEC                          N   N
   of PTLS not configured     |^600 (dsk)|        420 *  DISK420 RZ28     (C) DEC                          N   Y
   (marked with ^)            |          |        440 *  DISK440 RZ26     (C) DEC                          N   Y
4. Delete all unbounded       |          |        530 *  DISK530 RZ26     (C) DEC                          N   N
   devices (marked with *)    |          |        550 *  DISK550 RZ28     (C) DEC                          N   N
5. Return to main menu        |          |        620 *  DISK620 RZ28     (C) DEC                          N   N
                              |          |        640 *  DISK640 RZ26     (C) DEC                          N   Y
                              |          |
                              |          |
                              |          |
_________________________________________________________________________________________________________________________________


Enter menu choice (1,5) [5] ?
```

SMCS1-07

CFMENU has added information to the following fields:

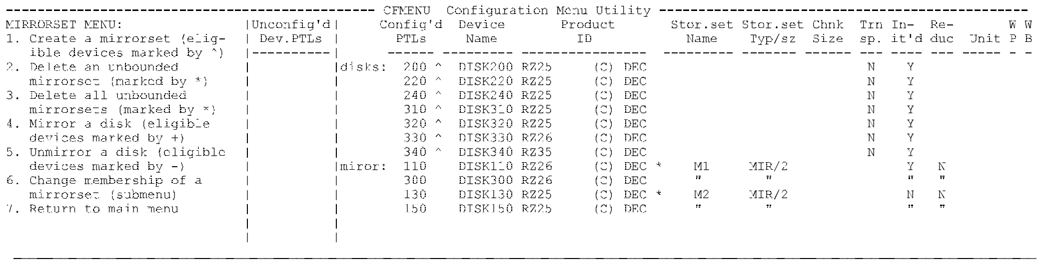
- **Config'd PTLs** – Configured PTLs. The program lists the PTL of each device added to your configuration.
- **Device Name** – CFMENU automatically assigns a name to each device. The name contains both the device type and PTL. You cannot override the automatic CFMENU naming convention.
- **Product ID** – Information identifying the device model appears here.
- **Trnsp** – Transportable. This field identifies whether or not the device is transportable.
- **Init'd** – Initialized. This field identifies whether or not the device is initialized.

After adding devices, return to the main menu.

8.5.4 Adding MIRRORSETS

Enter option 2 from the main menu to work with mirrorsets. From the mirrorset menu (see Figure 8-9), enter option 1 to add a mirrorset. CFMENU will prompt you for how many and which devices from the configured PTLs list you want to include in the mirrorset (two to six devices allowable). In Figure 8-9, two mirrorsets were created, one from disks at PTLs 110 and 300, and the other from disks at PTLs 130 and 150.

Figure 8–9 CFMENU Mirrorset Menu



```
----------------------------------------------- CFMENU  Configuration Menu Utility -----------------------------------------------
MIRRORSET MENU:               |Unconfig'd|     Config'd  Device       Product        Stor.set Stor.set Chnk  Trn In-  Re-        W W
1. Create a mirrorset (elig-  | Dev.PTLs |       PTLs     Name          ID             Name    Typ/sz  Size  sp. it'd duc   Unit P B
   ible devices marked by ^)  |----------|      ------ --------- ---------------- --------- ------- ------ --- ---- --- ----- - -
2. Delete an unbounded        |          |disks:  200 ^  DISK200 RZ25      (C) DEC                            N    Y
   mirrorset (marked by *)    |          |        220 ^  DISK220 RZ25      (C) DEC                            N    Y
3. Delete all unbounded       |          |        240 ^  DISK240 RZ25      (C) DEC                            N    Y
   mirrorsets (marked by *)   |          |        310 ^  DISK310 RZ25      (C) DEC                            N    Y
4. Mirror a disk (eligible    |          |        320 ^  DISK320 RZ25      (C) DEC                            N    Y
   devices marked by +)       |          |        330 ^  DISK330 RZ26      (C) DEC                            N    Y
5. Unmirror a disk (eligible  |          |        340 ^  DISK340 RZ35      (C) DEC                            N    Y
   devices marked by -)       |          |miror:  110    DISK110 RZ26      (C) DEC *    M1      MIR/2                  Y    N
6. Change membership of a     |          |        300    DISK300 RZ26      (C) DEC      "         "                    "    "
   mirrorset (submenu)        |          |        130    DISK130 RZ25      (C) DEC *    M2      MIR/2                  N    N
7. Return to main menu        |          |        150    DISK150 RZ25      (C) DEC      "         "                    "    "
                              |          |
                              |          |
```

SMCS1-27

CFMENU updates the following fields after adding a mirrorset:

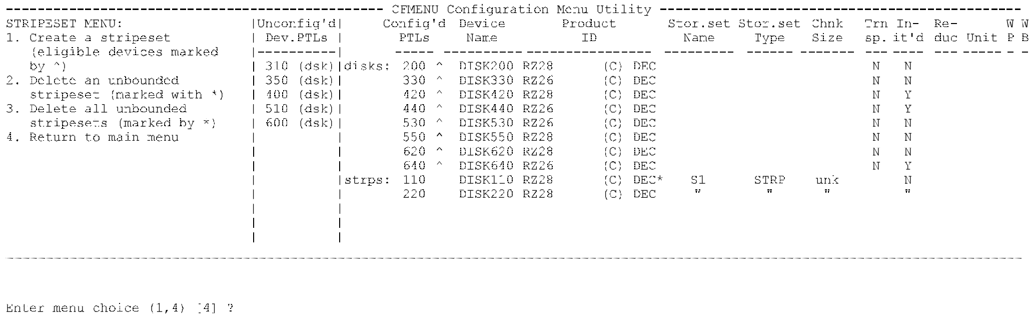
- **Stor.set Name –** Storageset name. CFMENU automatically assigns the name "Mx" to a mirrorset. You cannot alter this automatic CFMENU naming convention.
- **Stor.set Typ/Sz –** Storage type and size. This field will read "MIR" for storagesets that are mirrorsets and indicate the number of members in the mirrorset.

After adding mirrorsets, return to the main menu.

8.5.5 Adding Stripesets

Enter option 3 from the main menu to work with stripesets. From the stripeset menu (see Figure 8-10), enter option 1 to add a stripeset. CFMENU asks you how many and which devices from the configured PTLs list you wish to include in the stripeset (2-14 devices allowable). In Figure 8-10, a stripeset was created from disks at PTLs 110 and 220.

Figure 8–10 CFMENU Stripeset Menu



```
----------------------------------------------- CFMENU Configuration Menu Utility ---------------------------------------------
STRIPESET MENU:               |Unconfig'd|     Config'd  Device       Product        Stor.set Stor.set Chnk   Trn In-  Re-       W W
1. Create a stripeset         | Dev.PTLs |       PTLs     Name          ID             Name     Type   Size   sp. it'd duc  Unit P B
   (eligible devices marked   |----------|      -----  ---------- ---------------- --------- ------ ------- --- ---- --- ----- - -
   by ^)                      | 310 (dsk)|disks:  200 ^  DISK200 RZ28      (C) DEC                              N    N
2. Delete an unbounded        | 350 (dsk)|        330 ^  DISK330 RZ26      (C) DEC                              N    N
   stripeset (marked with *)  | 400 (dsk)|        420 ^  DISK420 RZ28      (C) DEC                              N    Y
3. Delete all unbounded       | 510 (dsk)|        440 ^  DISK440 RZ26      (C) DEC                              N    Y
   stripesets (marked by *)   | 600 (dsk)|        530 ^  DISK530 RZ26      (C) DEC                              N    N
4. Return to main menu        |          |        550 ^  DISK550 RZ28      (C) DEC                              N    N
                              |          |        620 ^  DISK620 RZ28      (C) DEC                              N    N
                              |          |        640 ^  DISK640 RZ26      (C) DEC                              N    Y
                              |          |strps:  110    DISK110 RZ28      (C) DEC*    S1       STRP    unk          N
                              |          |        220    DISK220 RZ28      (C) DEC     "         "       "           "
                              |          |
                              |          |
                              |          |
---------------------------------------------------------------------------------------------------------------------------------


Enter menu choice (1,4) [4] ?
```

SMCS1-03

CFMENU updates the following fields after adding a stripeset:

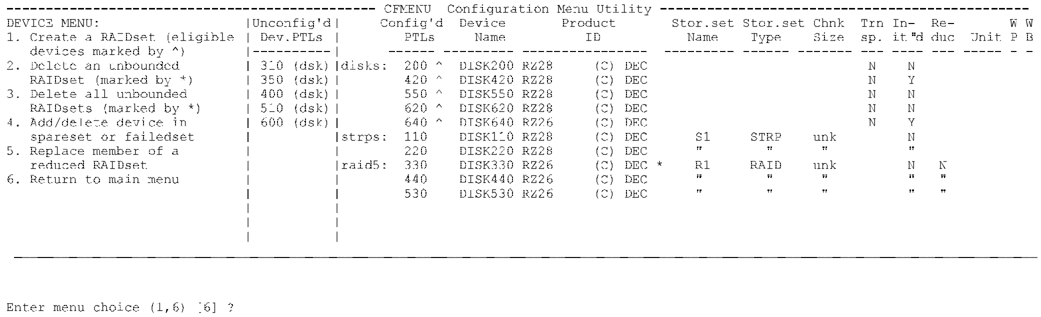
- **Stor.set Name** – Storageset name. CFMENU automatically assigns the name "Sx" to a stripeset. You cannot alter this automatic CFMENU naming convention.
- **Stor.set Type** – Storageset type. This field reads "STRP" for storagesets that are stripesets.
- **Chnk Size** – Chunk size. This field reads "unk" (unknown) until you initialize the stripeset.

After adding stripesets, return to the main menu.

8.5.6 Adding RAIDsets

Enter option 4 from the main menu to work with RAIDsets. From the RAIDset menu (see Figure 8-11), enter option 1 to add a RAIDset. CFMENU asks you how many and which devices from the configured PTLs list you wish to include in the RAIDset (3-14 devices allowable). In Figure 8-11, a RAIDset was created from disks at PTLs 330, 440, and 530.

Figure 8–11 CFMENU RAIDset Menu



```
----------------------------------------------- CFMENU  Configuration Menu Utility ------------------------------------------------
DEVICE MENU:                   |Unconfig'd|     Config'd  Device       Product       Stor.set Stor.set Chnk   Trn In-  Re-         W W
1. Create a RAIDset (eligible  | Dev.PTLs |        PTLs     Name          ID           Name    Type    Size   sp. it'd duc  Unit P B
   devices marked by ^)        |----------|      ------ --------- ----------------  --------- ------ ------- --- ---- --- ----- - -
2. Delete an unbounded         | 310 (dsk)|disks:  200 ^  DISK200 RZ28     (C) DEC                            N    N
   RAIDset (marked by *)       | 350 (dsk)|        420 ^  DISK420 RZ28     (C) DEC                            N    Y
3. Delete all unbounded        | 400 (dsk)|        550 ^  DISK550 RZ28     (C) DEC                            N    N
   RAIDsets (marked by *)      | 510 (dsk)|        620 ^  DISK620 RZ28     (C) DEC                            N    N
4. Add/delete device in        | 600 (dsk)|        640 ^  DISK640 RZ26     (C) DEC                            N    Y
   spareset or failedset       |          |strps:  110    DISK110 RZ28     (C) DEC      S1     STRP    unk         N
5. Replace member of a         |          |        220    DISK220 RZ28     (C) DEC      "       "       "          "
   reduced RAIDset             |          |raid5:  330    DISK330 RZ26     (C) DEC *    R1     RAID    unk         N   N
6. Return to main menu         |          |        440    DISK440 RZ26     (C) DEC      "       "       "          "   "
                               |          |        530    DISK530 RZ26     (C) DEC      "       "       "          "   "
                               |          |
                               |          |
                               |          |
 ___________________________________________________________________________________________________________________________________

Enter menu choice (1,6) [6] ?
```

SMCS1-08

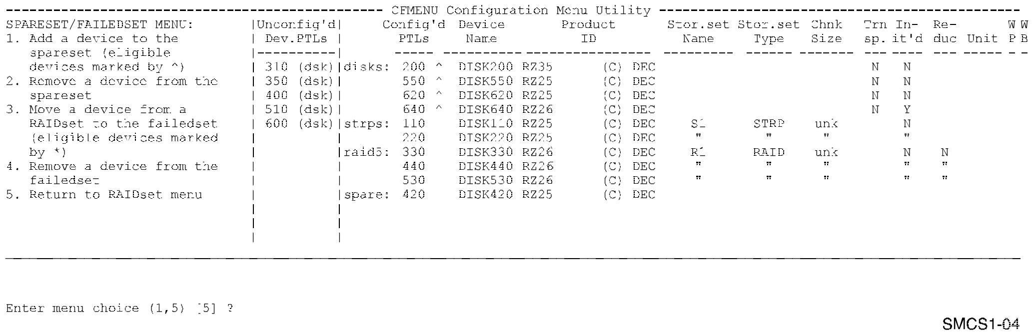
CFMENU updates the following fields after adding a RAIDset:

- **Stor.set Name** – Storageset name. CFMENU automatically assigns the name "Rx" to a RAIDset. You cannot alter this automatic CFMENU naming convention.
- **Stor.set Type** – Storageset type. This field reads "RAID" for storagesets that are RAIDsets.
- **Chnk Size** – Chunk size. This field reads "unk" (unknown) until you initialize the RAIDset.
- **Reduc** – Reduced. This field indicates whether or not a RAIDset is running reduced (missing one member).

8.5.7 Adding to Sparesets

Enter option 4 from the RAIDset menu to work with the sparesets and failedsets associated with RAIDsets. From the spareset/failedset menu (see Figure 8-12), you can, for example, enter option 1 to add a device to the spareset. CFMENU asks you which devices from the configured PTLs list that you wish to include in the spareset. In Figure 8-12, one device, PTL 420, was added to the spareset. After adding sparesets, return to the main menu via the RAIDset menu.

Figure 8–12 CFMENU Spareset/Failedset Menu



```
----------------------------------------------- CFMENU Configuration Menu Utility ---------------------------------------------
SPARESET/FAILEDSET MENU:       |Unconfig'd|     Config'd  Device         Product        Stor.set Stor.set Chnk    Trn In-  Re-       W W
1. Add a device to the         | Dev.PTLs |       PTLs     Name            ID              Name     Type    Size   sp. it'd duc  Unit P B
   spareset (eligible          |----------|       -----  ---------- ----------------  --------- ------ -------  --- ---- --- ----- - -
   devices marked by ^)        | 310 (dsk)|disks:  200 ^  DISK200 RZ35       (C) DEC                              N    N
2. Remove a device from the    | 350 (dsk)|        550 ^  DISK550 RZ25       (C) DEC                              N    N
   spareset                    | 400 (dsk)|        620 ^  DISK620 RZ25       (C) DEC                              N    N
3. Move a device from a        | 510 (dsk)|        640 ^  DISK640 RZ26       (C) DEC                              N    Y
   RAIDset to the failedset    | 600 (dsk)|strps:  110    DISK110 RZ25       (C) DEC    S1       STRP     unk          N
   (eligible devices marked    |          |        220    DISK220 RZ25       (C) DEC    "        "        "            "
   by *)                       |          |raid5:  330    DISK330 RZ26       (C) DEC    R1       RAID     unk          N    N
4. Remove a device from the    |          |        440    DISK440 RZ26       (C) DEC    "        "        "            "    "
   failedset                   |          |        530    DISK530 RZ26       (C) DEC    "        "        "            "    "
5. Return to RAIDset menu      |          |spare:  420    DISK420 RZ25       (C) DEC
                               |          |
                               |          |
                               |          |
________________________________________________________________________________________________________________________________


Enter menu choice (1,5) [5] ?
```

SMCS1-04

8.5.8 Initializing Containers

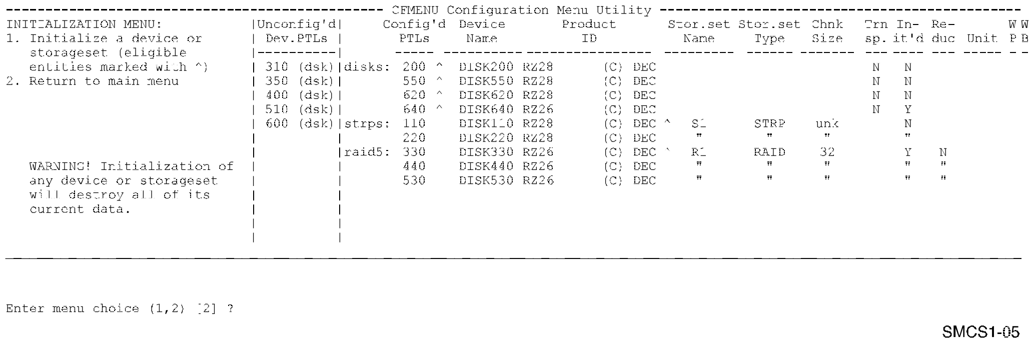
Enter option 5 from the main menu to initialize containers (devices or storagesets). From the initialization menu (see Figure 8-13), enter option 1. CFMENU asks you if you want to initialize each eligible container.

In addition, CFMENU asks you to decide on other operating qualifiers, depending on whether the container is a device, mirrorset, stripeset, or RAIDset. See the descriptions of the ADD or SET commands in Appendix B if you need help understanding the qualifiers.

In Figure 8-13, the RAIDset created earlier (R1) is initialized. Notice that R1 now has a chunksize of 32. The disk at PTL 640 has been initialized throughout this example, because CFMENU detected its condition when it was added to the configuration.

After initializing containers, return to the main menu.

Figure 8–13 CFMENU Initialization Menu



```
----------------------------------------------- CFMENU Configuration Menu Utility ---------------------------------------------
INITIALIZATION MENU:           |Unconfig'd|     Config'd  Device         Product        Stor.set Stor.set Chnk    Trn In-  Re-       W W
1. Initialize a device or      | Dev.PTLs |       PTLs     Name            ID              Name     Type    Size   sp. it'd duc  Unit P B
   storageset (eligible        |----------|       -----  ---------- ----------------  --------- ------ -------  --- ---- --- ----- - -
   entities marked with ^)     | 310 (dsk)|disks:  200 ^  DISK200 RZ28       (C) DEC                              N    N
2. Return to main menu         | 350 (dsk)|        550 ^  DISK550 RZ28       (C) DEC                              N    N
                               | 400 (dsk)|        620 ^  DISK620 RZ28       (C) DEC                              N    N
                               | 510 (dsk)|        640 ^  DISK640 RZ26       (C) DEC                              N    Y
                               | 600 (dsk)|strps:  110    DISK110 RZ28       (C) DEC ^  S1       STRP     unk          N
                               |          |        220    DISK220 RZ28       (C) DEC    "        "        "            "
                               |          |raid5:  330    DISK330 RZ26       (C) DEC ^  R1       RAID     32           Y    N
   WARNING! Initialization of  |          |        440    DISK440 RZ26       (C) DEC    "        "        "            "    "
   any device or storageset    |          |        530    DISK530 RZ26       (C) DEC    "        "        "            "    "
   will destroy all of its     |          |
   current data.               |          |
                               |          |
                               |          |
________________________________________________________________________________________________________________________________


Enter menu choice (1,2) [2] ?
```

SMCS1-05

8.5.9 Adding Units

Enter option 6 from the main menu to work with units. From the unit menu (see Figure 8-14), enter option 1 to add a unit. CFMENU asks you which initialized containers you wish to create units from.

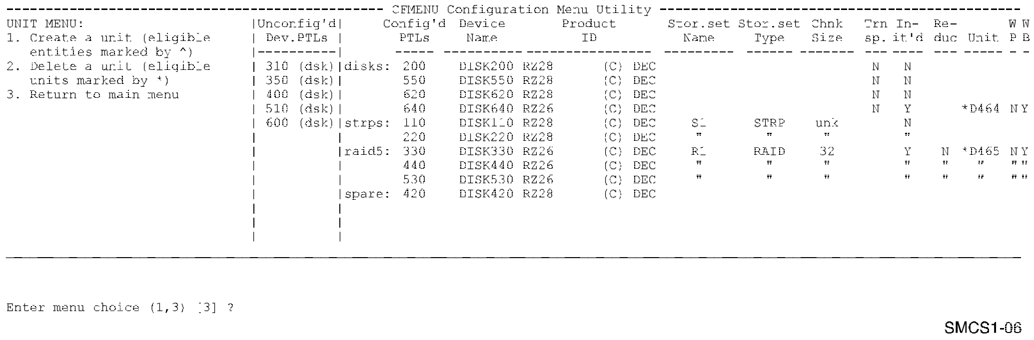
CFMENU also asks you to assign a unit number. (The program automatically assigns a "D" or "T" to the unit number when listing the unit, as shown in Figure 8-14.) In addition, CFMENU asks you to decide on other unit qualifiers. See the description of the ADD unit or SET unit commands in Appendix B if you need help understanding the qualifiers.

NOTE

Do not use CFMENU to modify the write-back cache setting for units. Accept the default setting when creating units.

In Figure 8-14, two units were created from RAIDset R1 and the disk at PTL 640.

Figure 8–14 CFMENU Unit Menu



```
------------------------------------------------ CFMENU Configuration Menu Utility ---------------------------------------------
UNIT MENU:                     |Unconfig'd|     Config'd Device        Product      Stor.set Stor.set Chnk   Trn In-  Re-      W W
1. Create a unit (eligible     | Dev.PTLs |       PTLs    Name           ID           Name     Type   Size   sp. it'd duc Unit P B
   entities marked by ^)       |----------|       -----  ------- ----------------   --------- ------ ------- --- ---- --- ---- - -
2. Delete a unit (eligible     | 310 (dsk)|disks: 200    DISK200 RZ28       (C) DEC                           N   N
   units marked by *)          | 350 (dsk)|       550    DISK550 RZ28       (C) DEC                           N   N
3. Return to main menu         | 400 (dsk)|       620    DISK620 RZ28       (C) DEC                           N   N
                               | 510 (dsk)|       640    DISK640 RZ26       (C) DEC                           N   Y      *D464 N Y
                               | 600 (dsk)|strps: 110    DISK110 RZ28       (C) DEC    S1      STRP    unk        N
                               |          |       220    DISK220 RZ28       (C) DEC     "       "       "         "
                               |          |raid5: 330    DISK330 RZ26       (C) DEC    R1      RAID    32         Y    N *D465 N Y
                               |          |       440    DISK440 RZ26       (C) DEC     "       "       "         "    "   "   " "
                               |          |       530    DISK530 RZ26       (C) DEC     "       "       "         "    "   "   " "
                               |          |spare: 420    DISK420 RZ28       (C) DEC
                               |          |
                               |          |
                               |          |
_________________________________________________________________________________________________________________________________

Enter menu choice (1,3) [3] ?
```

SMCS1-06

CFMENU updates the following fields after adding a unit:

- **Reduc**–Reduced. This field indicates whether or not a RAIDset is running reduced (missing one member).
- **Unit**–The unit number you assign appears here, preceded by the letter "D" or "T."
- **WP**–Write protect. This value indicates whether or not the unit is write protected.
- **WB**–Write-back. This value indicates whether or not the unit is set for write-back caching.

8.5.10 Terminal Setup

You can enter option 7 from the main menu to set the number of rows CFMENU displays. This feature is available primarily for terminals with the capability of displaying more than 24 rows.

8.5.11 Messages

This section lists the messages, other than the standard CLI messages, that CFMENU displays. However, most messages you see are those sent by the CLI, and are described in Appendix B.

```
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.

```
...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.

```
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.

```
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.

```
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 non-reduced RAIDset.

```
...Polling for unconfigured devices...
```

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

```
Port port# is blocked. No devices are 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 does not access devices on that bus.

```
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.

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

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

`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.

`There are no` *`devices/stripesets/RAIDsets/mirrorsets`* `eligible for deletion.`

Explanation: You chose to delete a device, stripeset, RAIDset, or mirrorset; but there are none that are eligible for deletion. A stripeset, RAIDset, or mirrorset may not be deleted if it is configured as a unit. A device may not be deleted if it is configured as a unit or if it is used in a storageset, spareset, or failedset.

`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.

`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.

`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 NO TRANSPORTABLE switch set and are configured as units or as part of a stripeset unit.

`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.

`There are no devices in the failedset.`

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

`There are no devices in the spareset.`

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

`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. 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.

```
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, 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.

```
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.

```
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.

```
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.

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

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

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

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

```
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.

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

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

8.5.12 Exiting CFMENU

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

8.6 CLCP Utility

The Code Load/Code Patch utility enables the user to perform two memory modification functions in the controller.

- Code loading-The user can upgrade the firmware in the controller's PCMCIA card via the maintenance port.
- Code patching-The user can alter memory locations in the controller's NVMEM, while the controller is active.

8.6.1 Invoking the CLCP Utility

The CLCP utility is invoked using the RUN command, via the CLI interface as follows:

```
CLI> 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] ?
```

The user selects the desired option, and the appropriate module of the utility executes, prompting for further inputs.

8.6.2 Code Loading

CAUTION

The write protect switch on the program card is shipped from the factory in the write protect position. Before loading or patching the program card, slide the write protect switch on the bottom right-hand end (when the label is facing up) of the card (this can be done with the card still inserted in its slot) to the left. This action shuts off the write protection and enables you to write the new information. When the card has been rewritten, remember to slide the switch back to the right (the write protect position).

The code load program is invoked from an external processor (typically a PC) connected to the maintenance terminal port. The processor must be configured to run the KERMIT terminal protocol at 19,200 baud, with 8 data bits, no parity, and one stop bit. When running the code load option of the CLCP utility from a personal computer, all screen savers should be disabled. Screen savers are terminate-and-stay-ready (TSR) programs and can interfere with the code load process.

To perform a code load operation, a file containing the new firmware image must be stored in the external processor. The file must be in binary image format. The binary firmware image file can be obtained from your service provider. See your service provider for further details on the distribution of firmware updates.

The user invokes the CLCP program from the CLI, via the external processor. The CLCP program then waits to be downloaded from the external processor, via the serial interface and the KERMIT file transfer protocol. The code load process takes approximately 27 minutes to successfully complete.

Once loaded into the SWXRC-04 controller's memory, the new image is written into the controller's PCMCIA card. The code load program then automatically initializes the controller to place the new controller firmware into effect.

8.6.2.1 Using the Code Load Utility

NOTE

If you run the Code Load option of the CLCP utility from a personal computer, disable all screen savers while the code load program is in process. Otherwise, they may interfere with the Code Load process.

Operate the code load utility as follows:

1. Load the binary firmware image file into the external processor, using file or network transfer utilities appropriate to the operating environment of the external processor.
2. Invoke the CLCP program and the select the code load option, as follows:

```
CLI> 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] ? 1
--------------------------------------------------------
```

```
This is the Code Load local program.  This program loads
a new firmware image on the controller program card.
Perform the file transfer from a computer that runs the
KERMIT file transfer protocol. Connect this computer via
a serial communication line  to the MMJ maintenance port
on the front of the controller module.

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

Perform the following steps before continuing:

* Access the new image file over the serial line from
the host computer.

* Configure KERMIT with the following parameters:

Terminal speed 19200 baud, eight bit, no parity, 1 stop
bit

It takes approximately 27 minutes to load the new image.


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

Start KERMIT now..
```

3. Using the KERMIT file transfer protocol from the external processor, download the binary firmware image file to the SWXRC-04 controller. This details of this action are specific to the operating environment of the external processor.
4. The code load program acknowledges the downloaded file, and writes the new firmware image to the PCMCIA program card memory.

```
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
```

8.6.3 Code Patching

The code patch module of the CLCP program can be run from either a maintenance terminal or a virtual host terminal. The user enters the appropriate patch information directly into the program, and the program places it into the controller's RAM memory. The patch becomes active after the first controller initialization.

The code patching utility allows more than one patch to be entered for a given firmware version. Each patch is associated with only one firmware version, and the code patch utility verifies the patch against the currently-installed firmware version. Some patches require the installation of previous patches, called dependent patches, before they can be installed. To identify it, each patch has a unique patch number. Operate the CLCP utility as follows:

1. Obtain the appropriate patch data for your controller's firmware version from your Digital representative.
2. Invoke the CLCP program, as follows:

```
CLI> 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. Select the desired code patch option, by entering an option number and pressing RETURN.

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.

8.6.3.1 Special Code Patch Considerations

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.
- Controllers in dual-redundant configurations must have the same patches applied, and patches must be entered into each controller separately.

8.6.3.2 Exit Option

Select this option to terminate the code patch program. Pressing Ctrl/Y at any time during the Code Patch program performs the same function.

8.6.3.3 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:

```
   CLI>    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 X25Z   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:       ? X25Z
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.
   CLI>
```

8.6.3.4 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.

```
   CLI> 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)
      ______       _________
       X25Z    -        1
Currently, 97% of the patch area is free.

Firmware Version of patch to delete ? X25Z
Patch Number to delete ? 1
The following patches have been selected for deletion:
```

```
 Firmware Version - Patch number(s)

     ______        _________

      X25Z    -       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] ?
```

8.6.3.5 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:

```
   CLI> 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)

   ______           _________
   V027     -        1
   X25Z     -        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 2.5 has had patches applied up to patch number three (3):

```
CLI> SHOW THIS_CONTROLLER
Controller:
     SWXRC-04 ZG33400026 Firmware V25Z-3, Hardware A-02
   .
   .
   .
```

8.6.3.6 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 CLI 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.

8.6.3.7 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.)

8.7 Firmware Licensing System

The firmware licensing system (FLS) enables or disables the licensed value-added software features of the SWXRC- 04 controller. You may use the FLS utility to perform the following tasks:

- Enable or disable optional functions for your controller
- 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.

IMPORTANT WRITE-BACK CACHE NOTE

If your system contains write-back cache modules, their batteries were completely charged at the factory. It is normal for the batteries to have discharged slightly in shipment. To ensure absolute data integrity, the advanced write-back cache and RAID features of your controller require fully-charged batteries to operate. These advanced features may not be available immediately after installation, until the batteries have had an opportunity to completely recharge.

8.7.1 Preenabled Licensed Features

If you purchased the licensed features for your controller, it may have been shipped with a preentered license key, to allow you to immediately use those features. To verify which features are enabled, use the SHOW THIS_CONTROLLER CLI command to display the state of the licensed features in your controller. You will also receive a copy of your license key, for future use in controlling the licensed features in your controller using the FLS utility.

8.7.2 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 terminal. These error indications are repeated at least once each hour while the unlicensed option remains enabled.

8.7.3 Disabling Options

You cannot disable an option if that option is currently in use. The following table lists the conditions under which you can disable an FLS option.

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

8.7.4 License Key

When you first run FLS, the license key is cleared. If you purchase 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.

8.7.5 Using the Menu

You can perform these operations from the FLS menu:

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

The following example shows the FLS main menu:

```
CLI> run fls
----------------------------------------------------------
Firmware Licensing System (FLS)
Option❶    State❷    License❸          Key❹
----------------------------------------------------------
RAID       ENABLED    ******INVALID!******  *none*
WBCA       ENABLED    VALID        ACME_WIDGET_CORP........
MIRR       DISABLED   INVALID      *none*
     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 RAID and WBCA are enabled. You may use any option that is enabled, regardless of whether you have a valid license key.

❸ **License** – RAID is running without a valid license. This status shows 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 purchase 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.

8.7.6 Example

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.

```
CLI> run fls
----------------------------------------------------------
Firmware Licensing System (FLS)
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 (0:2) [0] ? 2❶
Enter new WBCA key, including 8-character CRC, or enter 0
   to return to main menu: ACME_WIDGET_CORPVB8UWQ9C❷
*** License key verified ***
----------------------------------------------------------
Firmware Licensing System (FLS)
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 (0:2) [0] ? 2❸
*** WBCA enabled ***
----------------------------------------------------------
```

```
Firmware Licensing System (FLS)
Option     State        License         Key
----------------------------------------------------------
RAID       DISABLED     INVALID         *none*
WBCA       ENABLED      VALID❹      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] ?<Return>
FLS - Normal Termination
CLI>
```

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.

8.7.7 Messages

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

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.

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, this error is displayed if write-back caching is not enabled by FLS.

```
***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.

```
***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 8.7.3 for more information on disabling options.

8.8 Fault Management Utility

The Fault Management Utility (FMU) enables you to do the following:

- Control the spontaneous event logging and last failure logging displays
- Display controller last failure and memory system failure information. FMU can also provide a convenient way to review some error log information during your terminal session. Run FMU interactively during a terminal session by invoking it from the CLI prompt:

```
CLI> RUN FMU
```

FMU only interprets errors that occur after you install and run controller firmware containing FMU. In other words, FMU cannot search for "older" errors, and you cannot install FMU in an attempt to troubleshoot a preexisting error.

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

8.8.1 SET Command

The SET command controls the output from the FMU utility.

The SET command changes options for the two spontaneous displays:

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

Also, the SET command controls the options for the interactive displays available under the SHOW command.

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 do not appear when the terminal again becomes available. (The %EVL display is 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 interrupts the input.

Following the %EVL display, the CLI prompt and command input entered prior to the interruption are redisplayed. You can then complete the current command line (unless FMU reports another event).

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.

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.

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.

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--CLI> -- Instance Code: 01010302
```

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--CLI> --07-JUL-1994 07:44:48-- Instance Code:
   01010302

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

%EVL--CLI> 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.

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.

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.

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 qualifier(s) 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
```

8.8.2 SHOW Command

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

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 nonvolatile memory.

Qualifier	Description
MOST_RECENT	Displays the most recent last failure information.
ALL	Displays all the errors available in the buffer (up to four). Errors are displayed in descending order, starting with the most recent.
ENTRY *n*	Displays one of the entries from the buffer. 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 your service provider.

SHOW MEMORY_SYSTEM_FAILURE qualifier

This command interactively displays memory system failure information from any of the last failure entries stored in nonvolatile memory.

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 of	Displays all memory system failure information contained in any or all 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 that 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.

NOTE

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

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

SHOW PARAMETERS

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

8.8.3 DESCRIBE Command

The **describe** command displays decscriptive text for a numeric value contained in a particular event log field.

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>
```

8.8.4 EXIT Command

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

8.8.5 Examples

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

```
FMU> SHOW LAST_FAILURE ENTRY 4

Last Failure Entry: 4❶ Flags: 0007FA80❷

%FMU-01-Last Failure Event, Instance Code: 01010302❸

Power On Time: 0 Years, 41 Days, 4 Hours, 49 Minutes, 8
Seconds❹

❺Controller Model: SWXRC-04 Serial Number: ZG30355555
Hardware Version: 0000(00) A01 (01)

Controller Identifier:

Version: V25Z(25)❻

Informational Report

❸Instance Code 01010302 Description:

  An unrecoverable hardware detected fault occurred.

❼Last 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: 2❶ Flags: 0007FA8C❷

  Memory System Failure indicated

%FMU-14-Memory System Failure Event, Instance Code:
016E2D02❸

Power On Time: 0 Years, 41 Days, 21 Hours, 5 Minutes, 39
Seconds❹

❺Controller Model: SWXRC-04 Serial Number: ZG30355555
Hardware Version: 0000(00)

Controller Identifier:

Firmware

Reported via non-maskable interrupt
```

```
❽Memory 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
❸Instance Code 016E2D02 Description:
 The CACHEA0 DRAB detected a Nonexistent Memory Error
 condition during an I960 attempt to read CACHEA0 memory.
```

❶ **Entry**–A number representing the error position in the buffer.

❷ **Flags**–This value should be recorded and reported to your service provider.

❸ **Instance Code**–The instance code (and description) associated with this failure.

❹ **Power On Time**–The time of failure.

❺ **Controller**–Information identifying your controller.

❻ **Firmware Version**–Firmware version.

❼ **Last Failure Code**–Last failure code and description.

❽ **Memory Address**–The memory address, byte count, and register contents should be recorded and reported to your service provider.

```
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: SWXRC-04 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 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.

➊Current 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:     F    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: 00000000         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 your service provider has more information with which to troubleshoot the controller.

8.9 Crash Utility

CAUTION

The Crash utility causes a controller restart, and may result in the loss or corruption of data. Do not use the Crash Utility.

The crash utility was designed for use by Digital personnel in system checkout only. It is not intended as a user utility.

8.10 Volume Serial Number Utility

The volume serial number utility, CHVSN, enables the operator to view and change storage device volume serial numbers. Following is an example of the use of the CHVSN program:

The user invokes the CHVSN program and views the volume serial number:

```
CLI> RUN CHVSN
Device (port target lun)  [EXIT] ? 4 1 0
CHVSN: Volume Serial Number is 00000000 00000000
```

The user chooses to change the volume serial number for this volume and enters a new number:

```
Update CHVSN (Y/N) [N] ? Y
CHVSN: Volume Serial Number is 00012010 00580010
```

The user exits the CHVSN program:

```
Device (port target lun)  [EXIT] ? <Return>
```

CHVSN - Normal Termination

8.11 CLONE Utility

CLONE is a utility that partially automates the process of using the mirroring facility to create a snapshot copy of host unit data.

Using the mirroring facility to create a snapshot copy of host unit data is a multistep process. If your data snapshot needs are straightforward, CLONE can be used to relieve you of much of the chore of performing this common operation. Note, however, that 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 snapshot operation manually.

Although the operations involved in most data snapshot 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 circumstances will cause CLONE to cease operation:

- If the controller is reset or there is a power loss.
- If the host moves the unit being copied from one controller to the other.
- If you change the configuration of the unit while CLONE is running.
- If a disk device being used by CLONE fails.
- If the CLONE utility is aborted via ^Y or ^C.
- If the controller fails.

In most cases, CLONE can automate the operation entirely and will complete successfully without intervention. In those cases where it cannot complete, it stops at the point the error occurred. The CLI operations performed to the point of the error have been output and are displayed on your terminal screen.

Recovering partially completed CLONE operations requires you to be familiar with the data snapshot process as described in the examples that follow**.** The recovery strategy is to decide what follow-on operations will best meet your operational needs. You can continue the data snapshot operation to completion by manually issuing the remaining commands in the sequence. You can undo the actions that CLONE has taken by issuing appropriate CLI commands to reverse them. Alternatively, you can decide that some entirely different course of action is appropriate, and issue the appropriate CLI commands to proceed in that direction.

Each of the four CLONE examples that follow contain a detailed discussion of the typical sequence of operations that CLONE will perform for each of the legal target configurations. If a CLONE operation does not complete, use these template sequences and your knowledge of data snapshot operations in general to guide a manual recovery session.

Any unit created by CLONE has 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.

The following are examples for cloning a single device unit, cloning a stripeset, cloning a mirrorset, and finally, cloning a striped mirrorset.

Example 1: Cloning a Single Device Unit

```
CLI> RUN CLONE

Clone Local Program Invoked
Units available for cloning:
101
200
Enter unit to clone ? 101

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 ?
201
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 D201 C_M

D201 has been created. It is a clone of D101.

Clone - Normal Termination

CLI>
```

The following is the list of commands (in sequential order) you would use if you need to manually clone the single device unit from the preceding example

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*

   Wait for member NORMALIZATION
7. REDUCE *disk-device-name*
8. UNMIRROR disk-device-name
9. ADD MIRRORSET mirrorset-container-name disk-device-name
10. INITIALIZE container-name NODESTROY
11. ADD UNIT unit-number container-name

The following is an example of the specific manual commands used to clone the single device unit from the preceding example:

```
CLI>    SHOW DEVICES (displays the units available for cloning)
CLI>    SHOW STORAGESETS (displays the storagesets which are currently
        configured)
CLI>    MIRROR DISK110 C_M (converts the physical device DISK110 into a one
        member mirrorset and names the mirrorset C_M)
CLI>    SET C_M NOPOLICY (sets the replacement policy for mirrorset C_M to
        NOPOLICY so that spareset members are not automatically added to the
        mirrorset)
CLI>    SET C_M MEMBERSHIP=2 (sets the nominal membership of mirrorset
        C_M to two members)
CLI>    SET C_M REPLACE=DISK300 (places disk DISK300 into mirrorset C_M.
        The controller automatically starts copying data from the original member to
        the new member)
CLI>    REDUCE DISK300 (removes DISK300 from mirrorset C_M, after a copy
        from the existing mirrorset member to DISK300 has completed)
CLI>    UNMIRROR DISK110 (converts the one member mirrorset DISK110 back
        to a physical device)
CLI>    ADD MIRRORSET C_M DISK300 (creates a one-member mirrorset from
        DISK300 and names the mirrorset C_M)
CLI>    INITIALIZE C_M NODESTROY (initializes mirrorset C_M, but does not
        destroy any of the forced error metadata on the disk)
CLI>    ADD UNIT D797 C_M (creates a logical unit to the controller from the
        initialized mirrorset C_M and names the unit D797)
```

Example 2: Cloning a Stripeset

```
swxrc-04> show storagesets
Name   Storageset     Uses        Used by
-------------------------------------------
ST1    stripeset      DISK130     D799
                      DISK200
swxrc-04> 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

CLI>  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

CLI>  SHOW STRIPESETS

Name        Storageset         Uses        Used by

C_ST1       stripeset          C_MA        D798
                               C_MB
ST1         stripeset          DISK130     D799
                               DISK200

CLI>
```

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-name1 mirrorset-container-name2*
17. INITIALIZE *stripeset-container-name* NODESTROY CHUNK=
18. ADD UNIT *unit-name stripeset-container-name*

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

`CLI>` `SHOW DEVICES` (displays the units available for cloning)

`CLI>` `SHOW STORAGESETS` (displays the storagesets which are currently configured)

`CLI>` `MIRROR DISK130 C_MA` (converts the physical device DISK130 into a one member mirrorset and names the mirrorset C_MA)

`CLI>` `SET C_MA NOPOLICY` (sets the replacement policy for mirrorset C_MA to NOPOLICY so that spareset members are not automatically added to the mirrorset)

`CLI>` `SET C_MA MEMBERSHIP=2` (sets the nominal membership of mirrorset C_MA to two members)

`CLI>` `SET C_MA REPLACE=DISK220` (places disk DISK220 into mirrorset C_MA. The controller automatically starts copying data from the original member to the new member)

`CLI>` `MIRROR DISK200 C_MB` (converts the physical device DISK200 into a one member mirrorset and names the mirrorset C_MB)

```
CLI>   SET C_MB NOPOLICY (sets the replacement policy for mirrorset C_MB
       to NOPOLICY so that spareset members are not automatically added to the
       mirrorset)
CLI>   SET C_MB MEMBERSHIP=2 (sets the nominal membership of mirrorset
       C_MB to two members)
CLI>   SET C_MB REPLACE=DISK240 (places disk DISK240 into mirrorset
       C_MB.The controller automatically starts copying data from the original
       member to the new member)
```

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

```
CLI>   REDUCE DISK220 DISK240 (removes DISK220 and DISK240 from
       their respective mirrorsets, after the copy operations from the existing
       mirrorset members has completed)
CLI>   UNMIRROR DISK130 (converts the one member mirrorset DISK130
       back to a physical device)
CLI>   UNMIRROR DISK200 (converts the one member mirrorset DISK200
       back to a physical device)
CLI>   ADD MIRRORSET C_MA DISK220 (creates a one-member mirrorset
       from DISK220 and names the mirrorset (C_MA))
CLI>   ADD MIRRORSET C_MB DISK240 (creates a one-member mirrorset
       from DISK240 and names the mirrorset (C_MB))
CLI>   ADD STRIPESET C_ST1 C_MA C_MB (creates a stripeset called
       C_ST1 from the two newly-created  mirrorsets (C_MA) and (C_MB))
CLI>   INITIALIZE C_ST1 NODESTROY CHUNK=128 (Initializes stripeset
       C_ST1 with a chunksize of 128 blocks, but does not write new metadata
       onto the members)
CLI>   ADD UNIT D205 C_ST1 (creates a logical unit for the host from the
       initialized stripeset C_ST1 and names the unit D205)
CLI>   SHOW DEVICES (shows the device configuration after the clone is
       complete)
CLI>   SHOW STRIPESETS (shows the stripesets after the clone is complete)
```

Example 3: Cloning a Mirrorset

```
CLI> 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

CLI>

CLI> SHOW MIRRORSETS

Name         Storageset    Uses       Used by
--------------------------------------------
M1           mirrorset     DISK110    D411
                           DISK220

CLI>

CLI> RUN CLONE

Clone Local Program Invoked

Units available for cloning:

    411

Enter unit to clone ? 401

Enter the unit number which you want assigned to the new unit?
402

The new unit may be added using one of the following methods:

1. Clone will stall after all members have been copied. The
   user must then hit 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 D402 C_M1

D402 has been created. It is a clone of D401.

Clone - Normal Termination

CLI>

CLI> 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

CLI>

CLI> SHOW MIRRORSETS

Name     Storageset    Uses         Used by
------------------------------------------
C_M1     mirrorset     DISK310      D402
M1       mirrorset     DISK110      D401
                       DISK220

CLI>
```

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

The following is an example of the commands for cloning a mirrorset:

`CLI>` SHOW DEVICES (displays the units available for cloning)

`CLI>` SHOW MIRRORSETS (displays the mirrorsets which are currently configured)

`CLI>` SET M1 NOPOLICY (sets the replacement policy for mirrorset M1 to NOPOLICY so that spareset members are not automatically added to the mirrorset)

`CLI>` SET M1 MEMBERSHIP=3 (sets the nominal membership of mirrorset M1 to three members)

`CLI>` SET M1 REPLACE=DISK310 (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)

...wait for NORMALIZATION of M1...

`CLI>` REDUCE DISK310 (removes DISK310 from mirrorset M1 after the copy operation has completed)

`CLI>` ADD MIRRORSET C_M1 DISK310 (creates a mirrorset called C_M1 from DISK310)

`CLI>` INITIALIZE C_M1 NODESTROY CHUNK=128 (initializes mirrorset C_M1 with a chunksize of 128 blocks, but does not write new metadata onto the members)

`CLI>` ADD UNIT D499 C_M1 (creates a logical unit for the host from mirrorset C_M1 and names the unit D499)

`CLI>` SHOW DEVICES (shows the device configuration after the clone is complete)

`CLI>` SHOW MIRRORSETS (shows the mirrorsets after the clone is complete)

Example 4: Cloning a striped mirrorset

```
CLI> SHOW DEVICES
Name   Type   Port  Targ  Lun     Used by
------------------------------------------
DISK110disk    1     1    0        M1
DISK150disk    1     5    0
DISK220disk    2     2    0        M1
DISK310disk    3     1    0        M2
DISK350disk    3     5    0
DISK420disk    4     2    0        M2
DISK510disk    5     1    0        M3
DISK550disk    5     5    0
DISK620disk    6     2    0        M3


CLI> SHOW MIRRORSETS
Name    Storageset   Uses        Used by
------------------------------------------
M1      mirrorset    DISK110     ST1
                     DISK220
M2mirrorset          DISK310     ST1
                     DISK420
M3mirrorset          DISK510     ST1
                     DISK620
CLI>
CLI> SHOW STORAGESETS
Name    Storageset  Uses     Used by
------------------------------------------
ST1     stripeset   M1       D401
                    M2
                    M3
CLI>
CLI> RUN CLONE
Clone Local Program Invoked
Units available for cloning:
 401
Enter unit to clone ? 401
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?
402
```

```
The new unit may be added using one of the following methods:
 1. Clone will stall after all members have been copied. The
    user must then hit 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
Hit 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 D402 C_ST1
D402 has been created. It is a clone of D401.
Clone - Normal Termination
CLI>
CLI> 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
CLI>
CLI> 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
CLI>
```

```
CLI> SHOW STORAGESETS
Name     Storageset  Uses     Used by
----------------------------------------
C_ST1    stripeset   C_M1     D402
                     C_M2
                     C_M3
ST1      stripeset   M1       D401
                     M2
                     M3
CLI>
```

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

The following is an example of the commands for cloning striped mirrorsets:

```
CLI>    SHOW DEVICES (displays the devices available for cloning)
CLI>    SHOW MIRRORSETS (displays the configured mirrorsets)
CLI>    SHOW STORAGESETS (displays the configured storagesets)
CLI>    SET M1 NOPOLICY (sets the replacement policy for mirrorset M1 to
        NOPOLICY so that spareset members are not automatically added to the
        mirrorset)
CLI>    SET M1 MEMBERSHIP=3 (sets the nominal membership of mirrorset M1
        to three members)
CLI>    SET M1 REPLACE=DISK350 (places disk DISK350 into mirrorset M1.
        The controller automatically starts copying data  from existing NORMAL
        members to the new member)
CLI>    SET M2 NOPOLICY (sets the replacement policy for mirrorset M2 to
        NOPOLICY so that spareset members are not automatically added to the
        mirrorset)
CLI>    SET M2 MEMBERSHIP=3 (sets the nominal membership of mirrorset M2
        to three members)
CLI>    SET M2 REPLACE=DISK150 (places disk DISK150 into mirrorset M2.
        The controller automatically starts copying data from existing NORMAL
        members to the new member)
CLI>    SET M3 NOPOLICY (sets the replacement policy for mirrorset M3 to
        NOPOLICY so that spareset members are not automatically added to the
        mirrorset)
CLI>    SET M3 MEMBERSHIP=3 (sets the nominal membership of mirrorset M3
        to three members)
CLI>    SET M3 REPLACE=DISK550 (places disk DISK550 into mirrorset M3.
        The controller automatically starts copying data from existing NORMAL
        members to the new member)
```

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

```
CLI>    REDUCE DISK350 DISK150 DISK550 (removes DISK350, DISK150, and
        DISK550 from their respective mirrorsets, after the copy operations from the
        existing NORMAL mirrorset members has completed)
CLI>    ADD MIRRORSET C_M1  DISK350 (creates new one-member mirrorset
        C_M1 that contains DISK350)
CLI>    ADD MIRRORSET C_M2  DISK150 (creates new one-member mirrorset
        C_M2 that contains DISK150)
CLI>    ADD MIRRORSET C_M3  DISK550 (creates new one-member mirrorset
        C_M3 that contains DISK550)
```

`CLI>` ADD STRIPESET C_ST1 C_M1 C_M2 C_M3 (creates a stripeset called C_ST1 from the three new mirrorsets C_M1, C_M2, and C_M3)

`CLI>` INITIALIZE C_ST1 NODESTROY (initializes stripeset C_ST1 with a chunksize of 128 blocks, but does not write new metadata onto the members)

`CLI>` ADD UNIT D402 C_ST1 (creates a logical unit for the host from stripeset C_ST1 and names the unit D402)

`CLI>` SHOW DEVICES (shows the device configuration after the clone is complete)

`CLI>` SHOW MIRRORSETS (shows the mirrorsets after the clone is complete)

`CLI>` SHOW STORAGESETS (shows the storagesets after the clone is complete)

9

Removing and Replacing Field Replaceable Units

This chapter describes procedures to remove and replace Field Replaceable Units.

This chapter describes how to remove and replace or install the following field replaceable units (FRUs) in both dual-redundant and nonredundant configurations:

- Controller module (including its mounting bracket, OCP, and bezel)
- Cache module (including write-back cache batteries)
- Program card
- SCSI host cable
- SCSI device port cables
- Shelf Blowers
- Shelf Power supplies

CAUTION

Do not attempt to replace or repair components within FRUs or equipment damage may result. Use the controller fault indications and error logs to isolate to the FRU level only.

This chapter also discusses how to warm swap controllers and storage devices.

9.1 Servicing Controller Modules

Servicing a controller module involves several considerations:

- Diagnosing the controller
- Shutting down the controller
- Deciding what to replace
  - A nonredundant controller
  - One dual-redundant controller
  - Both dual-redundant controllers

9.1.1 Diagnosing the Controller

Following are some general items to consider when troubleshooting controller faults:

Generally, if the green OCP reset (//) button is lit continuously, the controller module needs replacing. However, you need to be as familiar as possible with the failure or reason for replacing the module. Be sure you have followed troubleshooting basics:

1. Make a note of all visual indicators (OCP, device LEDs, and/or error messages) available to you.
2. For surviving controllers in dual-redundant pairs, try entering the RESTART OTHER_CONTROLLER command. The surviving controller may be keeping its companion from operating.
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.

   Refer to Chapter 7 for detailed information about errors and repair actions.

CAUTION

Do not use a controller failure situation as an opportunity to move devices or otherwise reconfigure your subsystem. Doing so prevents the controller from communicating with its units once the fault is corrected.

NOTE

You should decide exactly what you are servicing (a nonredundant controller, one dual-redundant controller, or both dual-redundant controllers) before proceeding to the following sections, as each procedure varies and has different consequences.

9.1.2 Using the Controller SHUTDOWN Command to Turn Off Controller Power

If you want to turn off the power to your controller subsystem for any reason (such as a long holiday, system move, replacing a bad SCSI host cable, and so forth), make sure you turn off the power properly by using the following steps:

> **CAUTION**
>
> If the correct steps for turning off the power to an SWXRC-04 controller are not followed, there is a potential for loss of data that may exist on any devices connected to the array.

1. Use the proper procedures for shutting down your operating system if the host system is also going to be turned off.

   If the host is not going to be turned off, a shut down of the host system is not necessary, but use the proper operating system procedures to dismount any units that are accessed through the SWXRC-04 controllers.
2. When the dismount and/or the operating system shutdown procedures are complete, invoke the controller SHUTDOWN commands from the CLI.

> **CAUTION**
>
> Do not turn off the power to the controller subsystem until all shutdown procedures have successfully completed.

3. When you enter the controller SHUTDOWN command, do not specify any nondefault qualifiers. The default qualifiers allow discovery of any errors that might be present so that proper recovery procedures can be implemented before turning off the power to the system.
4. If you have a dual-redundant controller configuration, shutdown each controller one at a time. From a maintenance terminal connected to the maintenance port of one controller, use the following commands:

   ```
   CLI> SHUTDOWN OTHER_CONTROLLER
   CLI> SHUTDOWN THIS_CONTROLLER
   ```

   If you have a nonredundant controller configuration, only the SHUTDOWN THIS_CONTROLLER command is necessary.
5. Only when the controller SHUTDOWN command has successfully completed, may you turn off the power to the controller subsystem. If you are just shutting down the controller shelf and not the entire system, unplug the power supplies in the controller shelf.
6. If the controller configuration contains any devices or storagesets (RAIDsets, mirrorsets) that are write-back cache enabled, and the system is going to be turned off for an extended length of time, the batteries on the write-back cache modules will drain, causing the data in the cache modules to be lost.

After restoring power to the system, and restarting the controllers, check the battery status before accessing the RAIDset or mirrorset units from the operating system. The battery status must say "good," unless the CACHE_POLICY=B qualifier is set. Remember that if you have the CACHE_POLICY=B set, and you decide to use your RAIDset or mirrorsets, you risk losing data if power should be lost before the batteries have recharged. To verify the status of the batteries, enter the following commands:

```
CLI> SHOW THIS_CONTROLLER
```

or

```
CLI> Show OTHER_CONTROLLER
```

If the battery status is not "good," you will not be able to access the RAIDset or mirrorset units until the batteries have recharged unless you have the SET THIS_CONTROLLER or SET OTHER_CONTROLLER CACHE_POLICY=B qualifier set.

If you are not turning off the power and you want to quickly remove and replace *one* controller in a *dual-redundant* configuration, you may use the controller warm swap procedure (refer to Section 9.8.2), also called C_SWAP. This method provides the fastest, most transparent way of exchanging controllers with minimal system impact. Make sure the devices and storagesets for the failed controller fail over to the good controller before you start the C_SWAP program.

If you have a nonredundant controller configuration, use the controller SHUTDOWN command before removing the controller module from the controller shelf if the controller is still functioning.

If the controller in your nonredundant configuration has completely failed, simply remove it using the controller remove and replace procedures in this manual and replace it with a new controller module.

CAUTION

Make sure you keep a current copy of your configuration because you will need to set the initial parameters for the new controller. You also need it to set all of your device and storageset assignments. Without a copy of your configuration, you have no way to recreate the previous configuration.

See Appendix B for a complete description of the SHUTDOWN command and its qualifiers. Be sure to understand the consequences to data and devices when using any qualifiers.

9.1.3 Removing and Replacing a Nonredundant Controller

Nonredundant controllers in SWXSS-01 shelves are always installed in slot (SCSI ID) 7, the slot furthest from the SCSI device cable connectors. Nonredundant controllers in SWXSC-AA cabinet shelves may be installed in either slot (SCSI ID 6 or 7).

When you replace the controller module in a nonredundant configuration, device service is interrupted for the duration of the service cycle.

In effect, following any procedure to remove and replace an SWXRC-04 controller is a kind of "warm swapping." This is because other targets on the host SCSI bus remain unaffected. However, take care not to confuse simply removing and replacing an SWXRC-04 controller with the special warm swap procedure described in Section 9.8.2.

9.1.3.1 Tools Required

You need the following tools to remove or replace the controller module:

- ESD strap
- 3/32-inch Allen wrench
- Flat-head screwdriver
- Small flat-head screwdriver

9.1.3.2 Precautions

Refer to Chapter 1 for ESD, grounding, module handling, and program card handling guidelines.

Ground yourself to an appropriate place on the cabinet before servicing the controller module.

9.1.3.3 Module Removal

Use the following procedure to remove the controller module:

1. Access the controller modules in the cabinet.
2. Examine the green OCP reset (//) LED on the controller OCP. If the green LED stays lit continuously after troubleshooting (refer to Section 9.1.1), the controller has failed and is already shut down. Proceed to step 6.
3. If the controller is fully or partially functioning (green LED flashing), connect a maintenance terminal to its maintenance port, shown in Figure 9-1, and enter the following commands:

```
CLI> SHOW THIS_CONTROLLER FULL
CLI> SHOW DEVICES FULL
CLI> SHOW UNITS FULL
```

4. Record the output from the commands and keep it available for reference.

5. To shut down the controller, enter the following command:

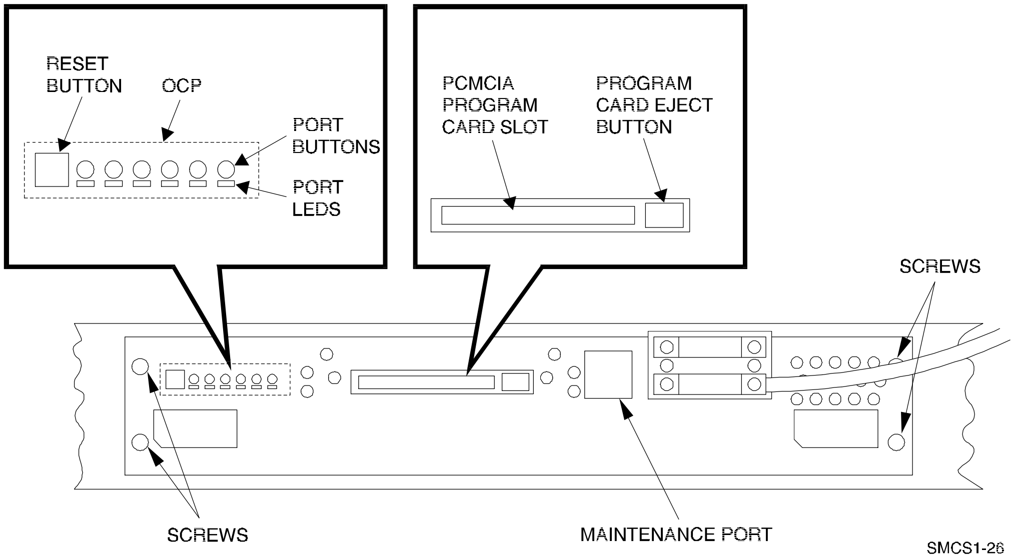
   CLI> **SHUTDOWN THIS_CONTROLLER**

   The green LED on the controller will light continuously when shut down completes.

> **CAUTION**
>
> *Never* remove a controller while it is still servicing devices. Doing so may destroy customer data.

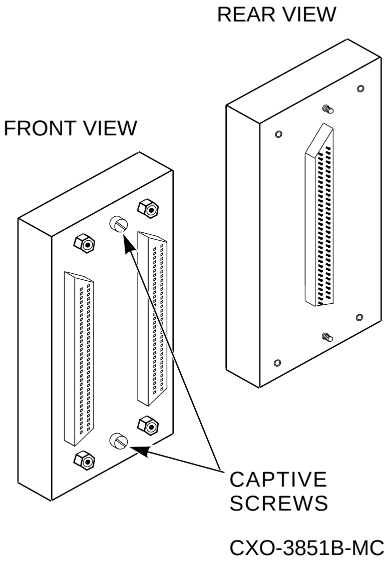
6. Unsnap and remove the ESD shield covering the program card by pulling out on the two retainer buttons.
7. Remove the program card by pushing the eject button on the controller OCP. Pull the card out and save it for use in the replacement controller module.

Figure 9–1 OCP Reset LED and Eject Button



With a small flat-head screwdriver, loosen the captive screws on the trilink connector (see Figure 9-2) and remove the trilink from the front of the controller. Do *not* remove cables or terminators from the trilink or you will interrupt the host SCSI bus. You will have to work around any SCSI cable or terminator connections when removing the trilink.

Figure 9–2 Trilink Controller



8. Remove the maintenance terminal cable (if attached).
9. Loosen the four mounting screws (refer to Figure 9-1) on each side of the front bezel with a flat-head screwdriver.
10. Use a gentle up-and-down rocking motion to loosen the module from the shelf backplane.
11. Slide the module out of the shelf (noting which rails the module was seated in) and place on an approved ESD work surface or mat.
12. If necessary, you may now remove the cache module as described in Section 9.2.4.3 or 9.2.1.4.

9.1.3.4 Module Replacement/Installation

Use the following procedure to replace or install the controller module:

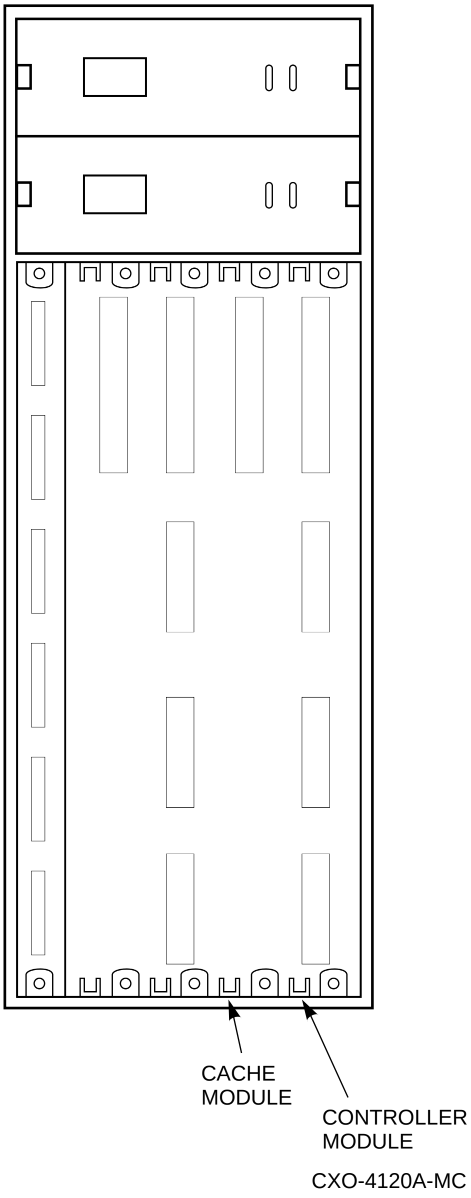
1. You should replace the cache module now, if you removed it. See Sections 9.2.4.4 and 9.2.1.5 for further information on replacing or installing the cache module.
2. Slide the controller module into the appropriate slot in the shelf. The slot arrangement of the SWXSS-01 shelf is shown in Figure 9-3.
3. Use a gentle up-and-down rocking motion to help seat the module into the backplane. Press firmly on the module until it is seated. Finally, press firmly once more to make sure the module is seated.
4. Tighten the four screws on the front bezel using a flat-head screwdriver.
5. Connect a maintenance terminal to the maintenance port of the new controller.
6. Press and hold the controller's green reset (//) button. Then insert the program card into the new controller. The program card eject button extends when the card is fully inserted.

7. Release the reset button to initialize the controller.

   If the controller initializes correctly, its green reset LED begins to flash at 1 Hz. If an error occurs during initialization, the OCP displays a code. If necessary, refer to Chapter 7 to analyze the code.

8. Snap the ESD shield into place over the program card.
9. Set initial controller parameters by following the steps in Section 9.1.3.5.

Figure 9–3 Controller Shelf Rails



10. *After setting initial parameters,* reconnect the trilink connector and host cables.
11. If you wish, you may disconnect the maintenance terminal. The terminal is not required for normal controller operation.
12. Close and lock the cabinet doors.

9.1.3.5 Restoring Initial Parameters

A new controller module has no initial parameters, so you must use the maintenance terminal to enter them. Refer to information that you recorded, at installation time or just prior to removing the controller (see Section 9.1.3.3), whichever is most current, for parameters. Be sure to set the parameters the same for the replacement controller as they were in the removed controller.

After installation of a nonredundant controller, use the CLI to define its parameters in the following order (from a maintenance terminal).

1. Enter the following command to set a valid controller ID:

   ```
   CLI> SET THIS_CONTROLLER ID=n
   ```

   where *n* is the SCSI target ID(s) (0-7).

   > **NOTE**
   >
   > Always restart the controller after setting the ID.

2. Restart the controller either by pressing the green reset (//) button, or by entering the following command:

   ```
   CLI> RESTART THIS_CONTROLLER
   ```

3. Enter the following command to verify the preceding parameters were set:

   ```
   CLI> SHOW THIS_CONTROLLER FULL
   ```

4. Connect the host port cable to the front of the controller.

   Connect the SCSI cable trilink connector to the front of the controller and tighten its captive screws with a small flat-head screwdriver. You must work around any SCSI cable or terminator connections when replacing the trilink. Do not remove cables or terminators from the trilink or you will interrupt the host SCSI bus.

To quickly configure devices on the controller, use either the CONFIG or CFMENU utility described in Chapter 8.

For manual configuration, the following steps add devices, storagesets, and logical units. Use the CLI to complete these steps so that the host recognizes the storage device. (These steps can be run from a virtual terminal.)

1. Add the physical devices by using the following command:

   ```
   CLI> ADD device-type device-name SCSI-location
   ```

   For example:

   ```
   CLI> ADD DISK DISK100 1 0 0
   ```

   where:

   device-type is the type of device to be added.

   device-name is the name to refer to that device. The name is referenced when creating units or storagesets.

   SCSI-location is the PTL for the device. When entering the PTL, at least one space must separate the port, target, and LUN.

2. Add the storagesets for the devices. Storagesets include stripesets, mirrorsets, and RAIDsets.

   See Appendix B for examples of adding storagesets. (If you do not desire storagesets in your configuration, skip this step.)

3. If restoring parameters after replacing a controller, skip this step and go to Step 4. Otherwise, enter the following command to initialize the containers (devices, storagesets, or both) prior to adding logical units to the configuration.

CAUTION

When restoring a container after swapping a controller, DO NOT run the INITIALIZE command. This is because the INITIALIZE command destroys all data on a container. See Appendix B for specific information on this command.

   ```
   CLI> INITIALIZE container-name
   ```

   where *container-name* is a device or storageset that will become a unit.

When initializing a single-device container:

- If NOTRANSPORTABLE (the default) was specified when the device was added, a small amount of disk space was made inaccessible to the host and used for metadata. The metadata is now initialized.
- If TRANSPORTABLE was specified, any metadata on the device is now destroyed. See Appendix B for details on metadata and when INITIALIZE is required.

4. Add the host-visible units by entering the following command:

   ```
   CLI> ADD UNIT logical-unit-number container-name
   ```

   where:

   - *logical-unit-number* is the unit number the host uses to access the device.
   - *container-name* identifies the device or the storageset.

5. Use the following commands to verify that your configuration matches the earlier, printed configuration:

```
CLI> SHOW DEVICES FULL
CLI> SHOW UNITS FULL
```

9.1.4 Removing and Replacing One Controller of a Dual-Redundant Pair

> **CAUTION**
>
> To perform the procedures in this section, at least one controller must be functioning.

To replace one controller in a dual-redundant configuration (or one at a time) while using the second controller to service devices, see the controller warm swap procedure in Section 9.8.2. Warm swap provides the fastest, most transparent way of exchanging controllers with minimal system impact and no down time.

When you replace one dual-redundant controller module using the instructions in this section, device service is interrupted for the duration of the service cycle.

9.1.4.1 Tools Required

You need the following tools to remove or replace the controller module:

- ESD strap
- 3/32-inch Allen wrench
- Flat-head screwdriver

9.1.4.2 Precautions

Refer to Chapter 1 for ESD, grounding, module handling, and program card handling guidelines. Ground yourself to an appropriate place in the cabinet before servicing the controller module.

9.1.4.3 Module Removal

Use the following procedure to remove the controller module:

1. If you have not done so already, unlock and open the cabinet doors to gain access to the controller modules.
2. Examine the green OCP reset (//) LED (refer to Figure 9-1) on both controllers. At least one green LED should not remain lit continuously after basic troubleshooting (refer to Section 9.1.1).

   If both green LEDs stay lit continuously, both controllers have failed. Refer to Section 9.1.5.

3. If the controller you are removing is still functioning (green LED flashing), connect a maintenance terminal to the controller. If the controller is not functioning, connect the maintenance terminal to the other controller.

   From the maintenance terminal, enter the following commands:

   ```
   CLI> SHOW THIS_CONTROLLER FULL
   CLI> SHOW DEVICES FULL
   CLI> SHOW UNITS FULL
   ```

   If the controller you are removing is not functioning and your maintenance terminal is connected to the other controller, type SHOW OTHER_CONTROLLER FULL in place of the first command, above.

4. Record the output from the commands and keep it available for reference.

> **CAUTION**
>
> Never remove a controller while it is still servicing devices. Doing so may destroy customer data.

5. If the controller you are removing is still functioning (green LED flashing) shut down the controller by typing:

   ```
   CLI> SHUTDOWN THIS_CONTROLLER
   ```

   The controller’s green LED will light continuously when shutdown completes.

6. Connect the maintenance terminal to the still functioning controller (the one you are *not* removing) and enter the following commands:

   ```
   CLI> SET NOFAILOVER
   CLI> SHUTDOWN THIS_CONTROLLER
   ```

   The controller’s green LED will light continuously when shutdown completes.

7. On the controller you are removing, unsnap and remove the ESD shield covering the program card.

8. Remove the program card by pushing the eject button (refer to Figure 9-1) next to the card. Pull the card out and save it for use in the replacement controller module.

   With a small flat-head screwdriver, loosen the captive screws on the trilink connector and remove the trilink from the front of the controller. You must work around any SCSI cable or terminator connections when removing the trilink. Do not remove cables or terminators from the trilink or you will interrupt the host SCSI bus. (If necessary for controller access, loosen the captive screws on the trilink connector and remove it from the front of the companion controller.)

9. Remove the maintenance terminal cable (if attached).

10. Loosen the four screws (refer to Figure 9-1) on each side of the front bezel with a flat-head screwdriver.

11. Use a gentle up-and-down rocking motion to loosen the module from the shelf backplane.
12. Slide the module out of the shelf (noting which rails the module was seated in) and place on an approved ESD work surface or mat.
13. If necessary, you may now remove the cache module as described in Section 9.2.4.3 or 9.2.1.4.

9.1.4.4 Module Replacement/Installation

Use the following procedure to replace the controller module:

1. Replace the cache module now, if you removed it. Refer to Section 9.2.4.4 or 9.2.1.5.
2. Slide the controller module into the shelf and into the same slot.
3. Use a gentle up-and-down rocking motion to help seat the module into the backplane. Press firmly on the module until it is seated. Finally, press firmly once more to make sure the module is seated.
4. Tighten the four screws on the front bezel using a flat-head screwdriver.
5. Restart the controller you did not remove by pressing and releasing the green reset (//) button on the controller operator control panel (OCP).

   Wait for the controller you did not remove to initialize.
6. Connect a maintenance terminal to the maintenance port of the controller you did *not* replace, and enter the following command:

   ```
   CLI> RESTART THIS_CONTROLLER
   ```

   Wait for the controller you did not remove to initialize.
7. Press and hold both controllers' green reset (//) buttons. Then insert the program card into the new controller. The program card eject button extends when the card is fully inserted.
8. Release both reset buttons.
9. Snap the ESD shield into place over the program card.
10. Restore initial controller parameters by following the steps in Section 9.1.4.5.
11. If you wish, you may disconnect the maintenance terminal. The terminal is not required for normal controller operation.
12. Close and lock the cabinet doors.

9.1.4.5 Restoring Initial Parameters

A new controller module has no initial parameters, so you must use a maintenance terminal to enter them. Refer to information recorded at installation time or just prior to removing the controller (see Section 9.1.4.3), whichever is most current, for parameters. Be sure to use the same parameters from the removed controller when installing a replacement. Follow these steps:

> **CAUTION**
>
> SET FAILOVER establishes controller-to-controller communication and copies configuration information. Always enter this command on one controller only. COPY=*configuration-source* specifies where the *good* configuration data are located. Never blindly specify SET FAILOVER. Know where your good configuration information resides before entering the command.

1. Using a maintenance terminal connected to the controller you did *not* replace, enter the following command to copy configuration information to the new controller:

   ```
   CLI> SET FAILOVER COPY=THIS_CONTROLLER
   ```

   Both controllers return to the dual-redundant configuration and restart after entering this command.

2. Enter the following commands to verify the preceding parameters were set.

   ```
   CLI> SHOW THIS_CONTROLLER FULL
   CLI> SHOW OTHER_CONTROLLER FULL
   ```

3. Connect the host port cables to the front of the controllers. Do not connect the controllers in a dual-redundant pair to separate, different host CPUs.

   Connect the SCSI cable trilink connector to the front of the controller and tighten its captive screws with a small flat-head screwdriver. You must work around any SCSI cable or terminator connections when replacing the trilink. Do not remove cables or terminators from the trilink or you will interrupt the host SCSI bus.

4. Use the following commands to verify your configuration matches the earlier, printed configuration before proceeding:

   ```
   CLI> SHOW DEVICES FULL
   CLI> SHOW UNITS FULL
   ```

9.1.5 Removing and Replacing Both Controllers in a Dual-Redundant Configuration

In the rare event that both controllers in your dual-redundant configuration fail, both controllers' green OCP reset (//) LEDs are lit continuously. You must replace both controller modules.

> **CAUTION**
>
> Simultaneously replacing both controllers in a dual- redundant configuration causes system down time for the duration of the service cycle. DIGITAL recommends only using this procedure if both controllers fail, or if your system is off line already for another reason.
>
> Otherwise, to replace both controllers, follow the steps in Section 9.1.4. Replace the controllers one at a time and maintain device service.

Use the following guidelines to simultaneously replace both controllers:

1. Examine the green OCP reset (//) LED on both controllers. Follow basic troubleshooting guidelines (refer to Section 9.1.1), if necessary.
2. If either of the controllers is fully or partially functioning, connect a terminal to the functioning controller and enter the following commands; otherwise, skip to Step 5:

```
CLI> SHOW THIS_CONTROLLER FULL
CLI> SHOW OTHER_CONTROLLER FULL
CLI> SHOW DEVICES FULL
CLI> SHOW UNITS FULL
```

3. Record the output from the commands and keep it available for reference.
4. Type the following commands to shut down the controllers:

```
CLI> SHUTDOWN OTHER_CONTROLLER
CLI> SHUTDOWN THIS_CONTROLLER
```

> **CAUTION**
>
> *Never* remove a controller while it is still servicing devices. Doing so may destroy customer data.

5. Remove both controllers by referring to the steps 6 through 12 in Section 9.1.3.3.
6. Replace the first of the controllers as if this were a nonredundant configuration (refer to Section 9.1.3.4).
7. Replace the second controller by following the dual-redundant procedure (refer to Section 9.1.4.4).

9.2 Servicing Cache Modules

The following procedures cover the servicing of cache modules.

9.2.1 Removing and Replacing Write-Back Cache Modules

WARNING

Service procedures described in this guide that involve removing and replacing the write-back cache must be performed only by qualified service personnel.

CAUTION

In general, you should never service a write-back cache module that contains unwritten cache data, or data loss may result. Always use the SHOW THIS_CONTROLLER or SHOW OTHER_CONTROLLER command to check for unwritten cache data. (Note that CLEAR_ERRORS UNWRITEABLE_ DATA may flush unwritten cache data.)

In situations where losing unwritten cache data does not matter, you may proceed with service. However, the unwritten cache data can remain in the module (needlessly occupying memory) until you enter the CLEAR_ERRORS UNWRITEABLE_DATA command for that module. This command erases the data and enables you to use the full capacity of the module.

See the following sections to service a write-back cache module.

9.2.1.1 Tools Required

You need the following tools to remove or replace the write-back cache module:

- ESD strap
- Nonconductive ESD mat
- 3/32-inch Allen wrench
- Flat-head screwdriver

9.2.1.2 Precautions

Refer to Chapter 1 for ESD, grounding, module handling, and program card handling guidelines.

Ground yourself to appropriate place in the cabinet before servicing the write-back cache module.

9.2.1.3 Swapping Cache Modules Between Controllers

Digital does not recommend the swapping of cache modules from one controller to another. Is it best to repair or replace a failed cache module. However, if you need to reconfigure your storage system, and you decide to move a write-back cache module from one controller to another, do the following steps in sequence:

1. Before removing the write-back cache module from its original location, enter the CLI command SHUTDOWN THIS_CONTROLLER. Do not specify the IMMEDIATE_SHUTDOWN or IGNORE_ERRORS qualifiers.
2. After removing the write-back cache module from the backplane, reinstall the battery disable jumper (W1) over its two pins, then remove the jumper and place it over only one of the two pins. This action clears all unwriteable data in cache memory.
3. Install the write-back cache module in its new location.

9.2.1.4 Module Removal

The controller module's front bezel covers the write- back cache module. Any time you service a write-back cache, you must remove the controller module. You must shut down controller(s) based on considerations of configuration, down time, and so on. Refer to Section 9.1.

WARNING

The write-back cache is a high energy module. Do not allow the write-back cache to contact any conductive surface, or injury and/or equipment damage may result.

CAUTION

The write-back cache batteries add weight to the module. Grasp the module firmly by the side where the batteries are seated any time you handle the module.

2. Use a gentle up-and-down rocking motion to loosen the module from the shelf backplane.
3. Slide the write-back cache module out of the shelf, noting which rails it was seated in, and place it on an approved nonconductive ESD mat.

9.2.1.5 Module Replacement/Installation

1. To replace the write-back cache module, its controller module must already be removed. (You should replace the write-back cache before reinstalling the controller module.)

> **WARNING**
>
> The write-back cache is a high energy module. Do not allow the write-back cache to contact any conductive surface, or injury and/or equipment damage may result.

> **CAUTION**
>
> The write-back cache batteries add weight to the module. Grasp the module firmly by the side where the batteries are seated any time you handle the module.

2. Slide the write-back cache module into the shelf.
3. Use a gentle up-and-down rocking motion to help seat the module into the backplane. Press firmly on the module until it is seated. Finally, press firmly once more to make sure the module is seated.
4. Replace the controller module by referencing Section 9.1.

9.2.2 Battery Removal

You may want to remove the batteries to reuse them on another cache module or because they have failed.

A console message is displayed when the write-back cache batteries are low (refer to *Chapter 7*). However, to check the battery status, you may enter the SHOW *controller* command. The battery status is "GOOD," "LOW," or "BAD."

The steps in this section explain how to remove the batteries.

> **WARNING**
>
> The write-back cache batteries are high energy devices. Do not allow the battery contacts to touch any conductive surface, or injury and/or equipment damage may result.

> **NOTE**
>
> Although service described in this guide is limited to removing and replacing batteries, it is mandatory that your service representative never repair the write-back cache module while the batteries are installed.

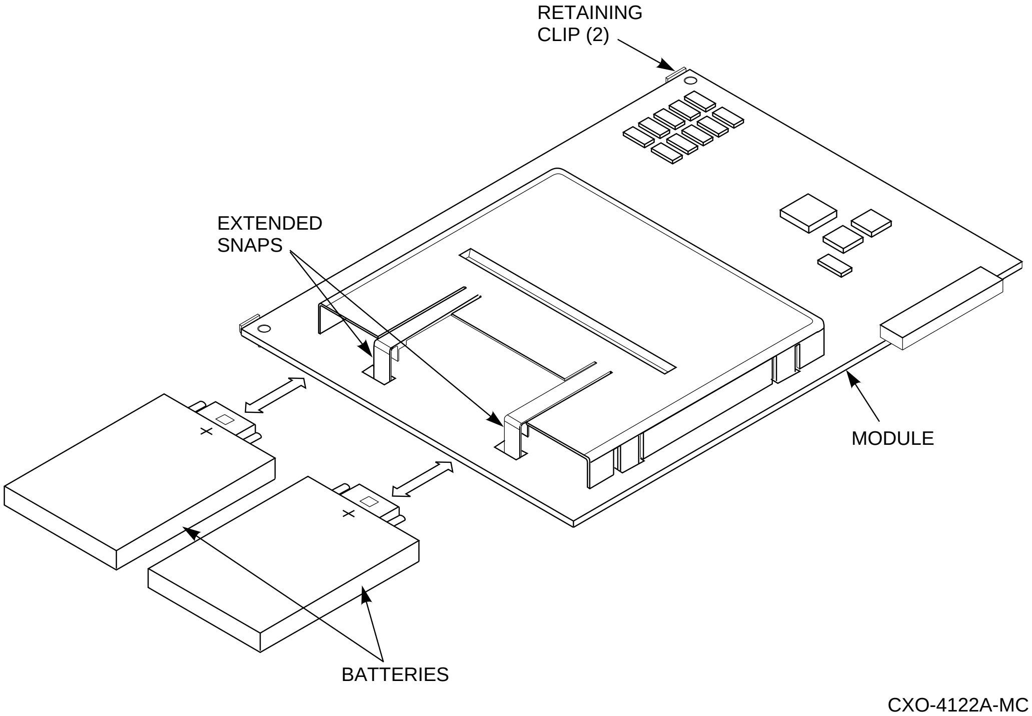
1. Locate the two write-back cache batteries seated under the plastic bracket at one end of the module.
2. Disengage and gently lift the extended snaps on the plastic bracket, and pull the batteries straight out of the module (see Figure 9-4). Place the batteries on an approved nonconductive ESD mat.

9.2.2.1 Battery Disposal

Dispose of batteries that are no longer capable of being recharged. Be aware that the write-back cache batteries contain lead. Therefore, follow disposal regulations applicable to their composition and marking in accordance with your local recycling laws.

Consult your service policies regarding hazardous materials handling for details. Do not simply discard spent batteries in the trash unless permitted by local regulations.

Figure 9–4 Write-Back Cache Batteries



9.2.2.2 Battery Replacement/Installation

WARNING

The write-back cache batteries are high energy devices. Do not allow the battery contacts to touch any conductive surface, or injury and/or equipment damage may result.

NOTE

Although service described in this guide is limited to removing and replacing batteries, it is mandatory that your service representative never repair the write-back cache module while the batteries are installed.

1. Disengage and gently lift the extended snaps on the plastic bracket and slide the batteries, positive (+) side up, straight into the module (refer to Figure 9-4).
2. Press the batteries firmly into the module so they are seated and the extended snaps close easily around the batteries.

   If you accidentally install a battery upside down, you will not be able to firmly seat the battery or close the retaining clip.

9.3 Servicing Program Cards

Whenever you remove a failed controller module (refer to Section 9.1), you remove the PCMCIA program card. However, there are times when you need to remove only the program card, such as when you install updated firmware.

You may remove one or both program cards from a dual-redundant configuration.

NOTE

When you update firmware, you must remove both program cards from a dual-redundant configuration. Furthermore, the two cards in a dual-redundant configuration must contain the same version of firmware.

Use the procedures in this section when you are removing and replacing only the program card.

9.3.1 Tools Required

- ESD Strap
- No other tools are required

9.3.2 Precautions

Refer to Chapter 1 for program card handling guidelines.

Ground yourself to the appropriate place on the cabinet before handling the program card.

9.3.3 Program Card Removal

Use the following procedure to remove the program card:

1. If you have not done so already, unlock and open the cabinet doors to gain access to the controller(s) and the program card(s).
2. Examine the green OCP reset (//) LED(s) on the controller(s). They should be flashing.

   If a green LED is lit continuously, its controller has failed. To service the controller, refer to Section 9.1.

   > **NOTE**
   >
   > You need not record configuration information; the configuration information is not lost when removing a program card.

3. Connect a maintenance terminal to the maintenance port of the controller(s) you are removing the program card from, and shut down the controller(s) by following the guidelines in Section 9.1.2.

   The green LED(s) should light continuously when shutdown completes.
4. Unsnap and remove the ESD shield(s) covering the program card(s) (refer to Figure 9-1).
5. Remove the program card(s) by pushing the eject button(s) (refer to Figure 9-1) next to the card(s).
6. Pull the card(s) out.
7. If you are updating firmware, follow the instructions included with your new firmware for used card return or disposal.

9.3.4 Program Card Replacement/Installation

Use the following procedure to replace the program card:

> **NOTE**
>
> If you are updating firmware, install your new program card(s) by following the instructions included with the card(s).Otherwise, you may use the following guidelines to replace the program card(s).

1. **Nonredundant configuration:**

   Press and hold the controller green OCP reset (//) button. Then insert the program card. The program card eject button extends when the card is fully inserted.

   Dual-redundant configuration:

   Press and hold *both* green reset buttons at the some time, *even if you are only replacing one of the cards*. Then insert the program card(s). The program card eject button extends when the card is fully inserted.
2. Release the reset button(s) to initialize the controller(s).

   If the controller(s) initialize correctly, the green reset LED(s) flash at 1 Hz. If an error occurs during initialization, the OCP(s) display a code. If necessary, refer to Chapter 7 to analyze any codes.
3. Snap the ESD shield(s) into place over the program card(s).
4. If you wish, you may disconnect the maintenance terminal. The terminal is not required for normal controller operation.
5. Close and lock the cabinet doors.

9.4 Servicing SCSI Host Cables

Servicing SCSI host cables (see Figure 9-5) may cause subsystem down time because the host path is disconnected from at least one controller for the duration of the procedure. (If you have a dual-redundant configuration, you should consider shutting down one controller and using the surviving controller to service devices while you replace the cable to its companion.) Use the procedures in this section when you are removing and replacing SCSI host cables.

> **CAUTION**
>
> Never leave active SCSI host buses unterminated during service. How you service your cables, and what devices you may leave running, depend upon your configuration.

The trilink connector may be considered part of the SCSI host cable during service.

9.4.1 Tools Required

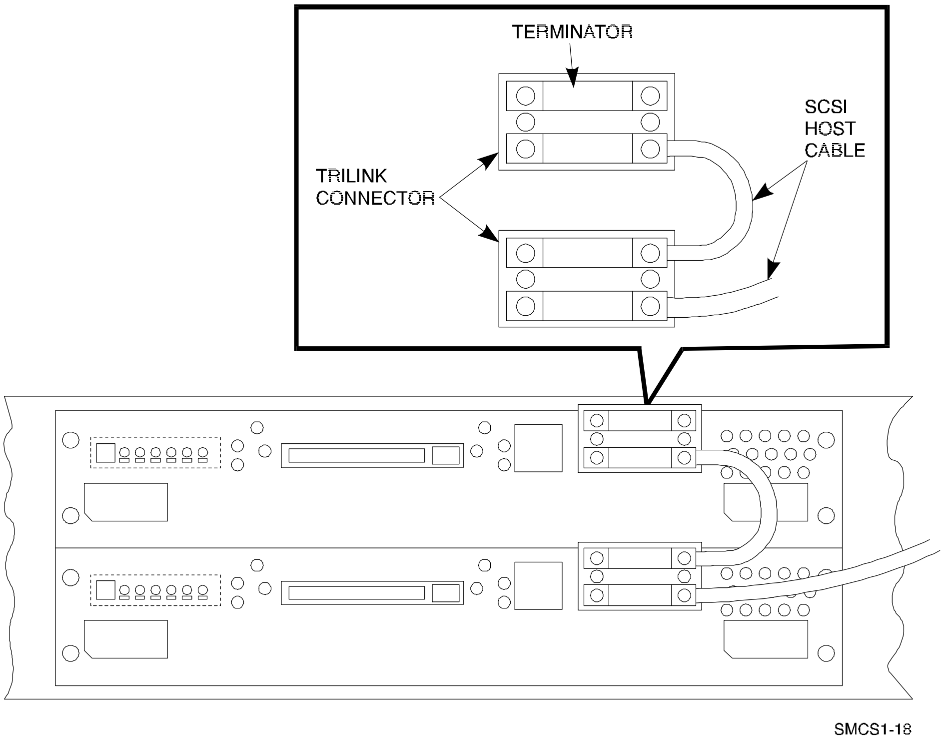
You need the following tools to remove or replace SCSI host cables:

- Tie wrap cutters
- Flat-head screwdriver

9.4.2 Precautions

Refer to Chapter 1 for general SCSI host cable handling guidelines.

Figure 9–5 SCSI Host Cables



9.4.3 Cable Removal

Use the following procedure to remove SCSI host cables:

1. Disconnect the SCSI host cable from the host or other device (the device at the other end of the cable from the controller).
2. If necessary to access the SWXRC-04 controller, unlock and open the cabinet to gain access to the cables.
3. Loosen the captive screws on the SCSI host cable where it attaches to the trilink connector on the front of the controller, and disconnect the cable.
4. Remove the SCSI host cable from the cabinet, cutting tie wraps as necessary.

5. (Optional) Loosen captive screws and remove the terminator or secondary SCSI host cable attached to the trilink connector.
6. (Optional) Loosen captive screws and remove the trilink connector from the front of the controller.

9.4.4 Cable Replacement/Installation

Use the following procedure to replace SCSI host cables:

1. (Optional) Attach the trilink connector to the front of the controller and tighten its captive screws.
2. Position and route the SCSI host cable within the cabinet.
3. Connect the SCSI host cable to the trilink connector on the front of the controller, and tighten the captive screws on the SCSI host cable connector.
4. (Optional) Connect and tighten captive screws for the terminator or secondary SCSI host cable (at the open connection of the trilink connector).
5. Install any tie wraps as necessary to hold the SCSI host cable in place.
6. Close and lock the cabinet doors.
7. Connect the other end of the cable to the appropriate device on the bus, removing terminators as necessary.

9.5 Servicing SCSI Device Port Cables

Servicing SCSI device port cables causes subsystem down time because you must remove devices to access SCSI connectors on the SWXSS-01 (controller) and SWXSS-02 (device) shelf backplanes.

NOTE

If the desired cable connects to a device shelf in the lower part of a cabinet, it may be easier to remove the device shelf rather than attempt this procedure with the shelf installed. Refer to the StorageWorks Solutions Shelf and SBB User's Guide for procedures to remove a device shelf and for correct SCSI cable lengths.

9.5.1 Tools Required

You need the following tools to remove or replace device port cables:

- ESD strap
- 3/32-inch Allen wrench
- Flat-head screwdriver

9.5.2 Precautions

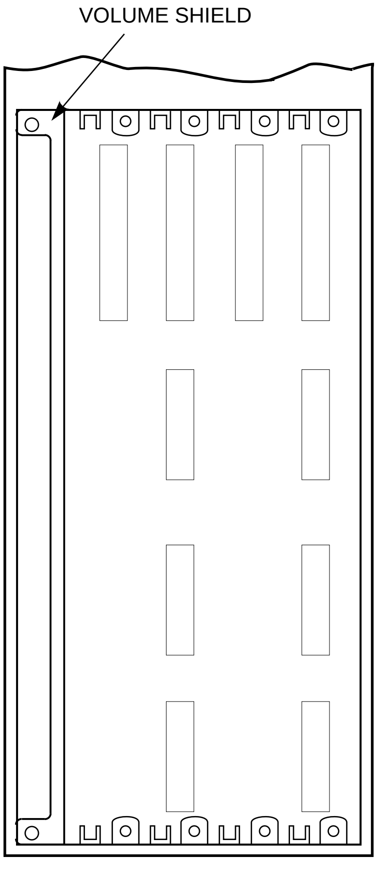
Refer to Chapter 1 for ESD, grounding, module handling, and cable handling guidelines.

9.5.3 Cable Removal

Use the following procedure to remove device port cables:

1. Unlock and open the cabinet to gain access to the cables.
2. To gain access to the cables in the controller shelf, remove the controller(s) and cache module(s) using the procedures described in Sections 9.1 and 9.2.
3. Using a flat-head screwdriver, loosen the two captive screws on each side of the volume shield, and remove the shield (see Figure 9-6).

Figure 9–6 Volume Shield



CXO-4176A-MC

4. Remove the cable from the SWXSS-01 (controller) shelf backplane by pinching the cable connector side clips and disconnecting the cable.

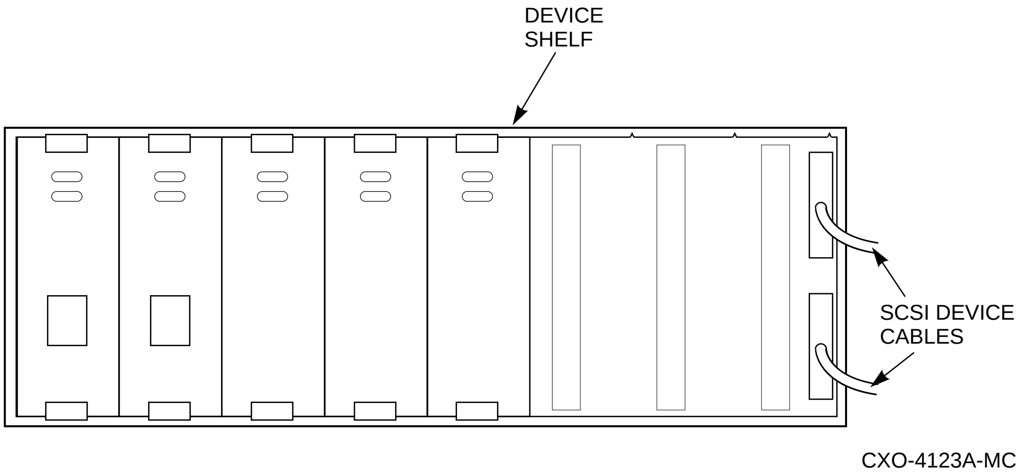
CAUTION

DIGITAL recommends labeling devices to indicate what slot they were removed from. If SBBs are removed and then returned to a different slot, customer data may be destroyed.

Let disk drives spin down for at least 30 seconds prior to removing them from the device shelf. Gyroscopic motion from a spinning disk may cause you to drop and damage the SBB.

5. Remove any SBBs necessary to allow access to the SCSI cables in the device shelf, as shown in Figure 9-7. (Press down on the two SBB mounting tabs to release it from the shelf, and pull the device straight out.)

Figure 9–7 SCSI Device Cables



6. Remove the cable from the SWXSS-02 (device) shelf backplane by pinching the cable connector side clips and disconnecting the cable.

9.5.4 Cable Replacement/Installation

CAUTION

Be very careful when inserting cable connectors into connectors within the SWXSS-01 and SWXSS-02 shelves. Inserting a poorly aligned cable connector can damage the shelf connector.

You must replace the entire shelf if its connectors are damaged.

Use the following procedure to replace device port cables:

1. For the device shelf connector, gently slide the cable connector in from one side to the other, and rock the connector from top to bottom to seat it.
2. Listen for the connector to snap into place.
3. For the controller shelf connector, *gently* slide the cable connector in from one side to the other, and rock the connector from top to bottom to seat it.
4. Listen for the connector to snap into place.

CAUTION

Return a device to the slot from which it was removed. If SBBs are removed and then returned to a different slot, *customer data may be destroyed.*

5. Insert the SBBs into the device shelf *making sure that all SBBs are returned to their original slots.*

   The SBB mounting tabs snap into place as the SBBs are locked into the shelf.
6. Replace the volume shield in the controller shelf and tighten the captive screws finger tight using a flat-head screwdriver (refer to Figure 9-6).
7. Replace the cache module(s) and controller(s) using the procedures described in Sections 9.1 and 9.2.
8. Close and lock the cabinet doors.

9.6 Servicing Shelf Blowers

WARNING

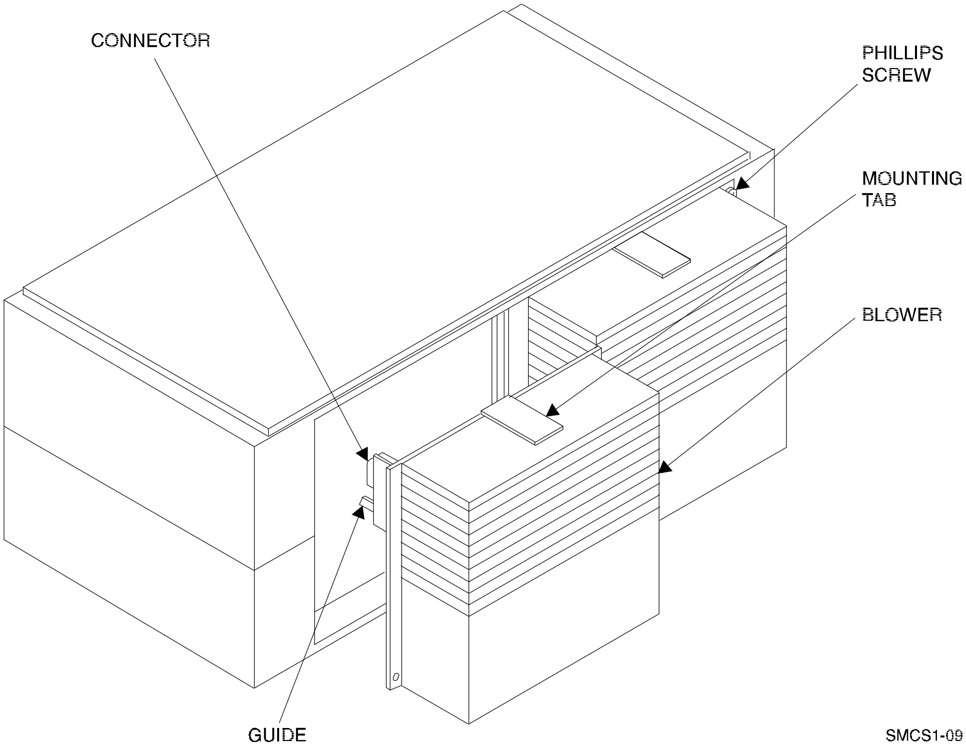
Service procedures described in this guide that involve shelf blower removal or access to the rear of the shelf must be performed only by qualified service personnel.

StorageWorks shelves have rear-mounted blowers that cool the controllers and storage devices (see Figure 9-8). Connectors on the shelf backplane provide +12 Vdc power to operate them.

When a blower in a SWXSS-series shelf fails, the shelf status (upper) LED on the power SBB turns off, and an error message is passed to the controller or host.

When a blower in a SWXSC-AA cabinet shelf fails, the condition is detected by the EMU and displayed on the EMU control panel.

Figure 9–8 Replacing a Blower



As long as one blower in an HA-000 or SWXSS-02 shelf is operating, there is sufficient air flow to prevent an overtemperature condition. If both blowers fail, the shelf can overheat in as little as 60 seconds. The SWXSC-AA cabinet can operate indefinitely with one failed blower.

9.6.1 Tools Required

You need the following tools to remove or replace the blower:

- Phillips screwdriver (#2)

9.6.2 Precautions

Refer to Chapter 1 for safety guidelines.

9.6.3 Blower Removal

> **WARNING**
>
> To reduce the risk of electrical energy hazard, disconnect the power cables from the shelf power supplies before removing shelf blower assemblies or performing service in the backplane area.

Use the following procedure to remove a blower:

1. Unlock and open the cabinet doors to gain access to the blowers. For SWXSC-AA cabinets, locate the dual-speed blowers on the rear of the cabinet.
2. If you cannot access the rear of the shelf, remove its SCSI device cables as described in Section 9.5. Then remove the shelf as described in the StorageWorks Solutions Shelf and SBB User's Guide.
3. Disconnect the power cables from the shelf power SBBs. The primary power supply cord is black. The secondary power supply cord is gray.
4. Use a Phillips screwdriver to remove the safety screw in the upper right corner or lower left corner of the blower. (This step is unnecessary for SWXSC-AA cabinets.)
5. Press the upper and lower blower mounting tabs together to release the blower.
6. Pull the blower straight out to disconnect it from the shelf power connector.

9.6.4 Blower Replacement/Installation

> **WARNING**
>
> To reduce the risk of electrical energy hazard, disconnect the power cables from the shelf power supplies before replacing shelf blower assemblies or performing service in the backplane area.

Use the following procedure to replace a blower:

1. Align the replacement blower connector and push the blower straight in, making sure it is fully seated and that both mounting tabs lock in place.
2. Replace the safety screw in the corner of the blower using a Phillips screwdriver. (This step is unnecessary for SWXSC-AA cabinets.)

3. If you had to remove the shelf to access the blowers, replace the shelf as described in the StorageWorks Solutions Shelf and SBB User's Guide. Then replace its SCSI device cables as described in Section 9.5.
4. Connect the shelf power cables and verify that the shelf and all SBBs are operating properly.

> **NOTE**
>
> If the upper power supply LED (shelf status) does not come on and all the shelf power supplies are operating, the second blower may have failed or the wrong blower may have been replaced.

5. Close and lock the cabinet doors.

9.7 Servicing Shelf Power Supplies

There are two methods for replacing shelf power supply SBBs: **hot swap** and **cold swap**.

- Use hot swap to replace a power supply *only* when there are redundant power supplies in a shelf. Hot swap enables you to remove defective power supplies while the other supplies furnish power.

> **NOTE**
>
> Hot swap does not disable the shelf or its contents.

- Use cold swap when the power is off to the shelf for some reason or when there are no redundant power supplies. Should this occur in an SWXSC-AA cabinet shelf or SWXSS-01 controller shelf, the controller, cache module, and all associated SCSI buses are disabled until power is restored. On a SWXSS-02 device shelf, those particular devices are disabled, though their controller still services devices on other shelves.

9.7.1 Tools Required

None required.

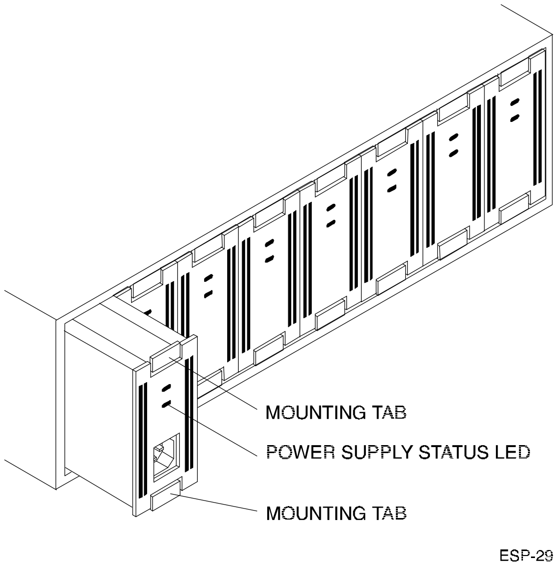
9.7.2 Precautions

Refer to Chapter 1 for safety guidelines.

9.7.3 Power Supply Removal

Use the following procedure to remove a power supply (see Figure 9-9):

Figure 9–9 Power Supply Removal



> **NOTE**
>
> The cold swap procedure is identical to hot swap, except you should shut down the shelf contents (devices or controllers) before removing the power supply.

1. Unlock and open the cabinet doors to gain access to the power supplies.
2. Make sure the power status (lower) LED on the power supply is off.
3. Unplug the power supply.
4. Press the two mounting tabs together to release the power supply from the shelf.

> **CAUTION**
>
> The power supply is relatively heavy and can be damaged if dropped. Always use *both* hands to fully support the power supply during removal.

5. Use both hands to pull the power supply out of the shelf.

9.7.4 Power Supply Replacement/Installation

Use the following procedure to replace a power supply (refer to Figure 9-9):

> **CAUTION**
>
> The power supply is relatively heavy and can be damaged if dropped. Always use both hands to fully support the power supply during replacement.

1. Hold the power supply in both hands and firmly push it into the shelf until you hear the mounting tabs snap into place.
2. Plug the power cord back into the power supply.
3. Observe the power and shelf status LEDs to make sure both turn on. If both LEDs do not turn on, refer to Chapter 7 for troubleshooting basics.
4. Close and lock the cabinet doors.

9.8 Warm Swapping Storage Subsystem Components

Industry definitions of cold, warm, and hot swap may vary considerably. For purposes of the StorageWorks family of products, these terms are defined as follows:

- **Cold swap:** a product replacement method where all system power and activity must be totally removed for the duration of service. This method is used when conditions preclude the use of the warm swap or hot swap methods.
- **Warm swap**: a replacement method where a product can be added, removed, or replaced while system power is on, but while some system activity is momentarily suspended during the procedure.
- **Hot swap**: a replacement method where the system remains powered on and fully active during product service. The product being removed or installed is the only item that cannot perform operations during this process.

When you warm swap a storage SBB or a controller, you quickly and safely remove the hardware and install a replacement. Warm swap is possible without taking your controllers out of service or adversely affecting activity on the rest of the subsystem. Using warm swap also preserves data integrity.

> **NOTE**
>
> Warm swap is not applicable to service on unpowered StorageWorks shelves. Do not attempt to execute warm swap on an unpowered shelf.

9.8.1 Warm Swapping Device SBBs

Device warm swap involves quickly removing and replacing the disk drive or other storage SBB. You can safely remove SBBs without taking your system or controller off line. However, before removing a device, either the controller or the operator must determine that the swap is necessary. You may also use the SBB warm swap procedure to add a device to an empty shelf slot.

- The controller determines that a device is bad by trying to access the device, receiving no response from the device, or detecting excessive errors from the device.
- The operator decides to remove a device by examining the OCP codes, the SBB LEDs, system messages, or system error log information.

NOTE

Most devices can be warm swapped; however, see your firmware release notes for restrictions.

9.8.1.1 Tools Required

None required.

9.8.1.2 Precautions

Refer to Chapter 1 for safety guidelines.

9.8.1.3 Device Removal

CAUTIONS

Warm swap supports removal and replacement of only one SBB at a time. Should another SBB need to be swapped, you must repeat the entire warm swap procedure.

You must follow steps in this section in their exact order so that the following is ensured:

Preserve data integrity (especially for devices with older SCSI interface designs).

Reduce chances of making a port unusable for a long period, which can render several devices inaccessible.

Prevent the controller from performing unpredictably.

Use the following procedure to remove a device:

1. For all configurations except those using RAIDsets, you must unmount the device from the host before proceeding.

   Refer to your operating system documentation for procedures necessary for dismounting a device.

   RAIDsets that are not already running as reduced automatically adjust to the removal of a device (the RAIDset goes reduced). In this case there is no need to dismount the suspect device. However, you must dismount the device if the RAIDset is already reduced.

2. Unlock and open the cabinet doors to gain access to the device SBBs.
3. Quiesce the SBB's port by pressing and holding the controller port button for the SBB. Continue holding the button until all amber OCP LEDs light.

> **NOTE**
>
> Only one port may be quiesced at any time.
>
> If the button is not held long enough, or multiple buttons are pushed in quick succession, all buttons are ignored (no ports are quiesced). You must press and hold the button again to quiesce the port.

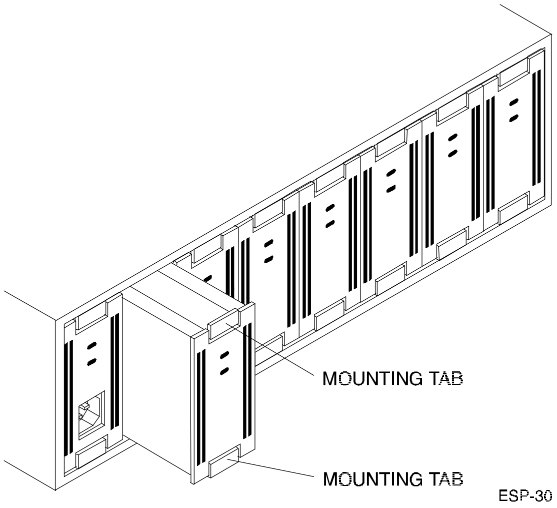
4. Wait until the chosen port LED flashes alternately with the other port LEDs (this indicates I/O has stopped). The alternating pattern flashes for approximately 30 seconds, during which you may remove (and, time permitting, replace) the SBB.

   If the pattern does not appear after a minute or two, another shelf is asserting a fault signal that prevents any quiesce function on this controller. To correct the problem, you must locate the suspect shelf and do one of three things:

   - Remove all devices from the shelf.
   - Disconnect the shelf's SCSI device cables (refer to Section 9.5).
   - Repair/replace the shelf power supply (refer to Section 9.7).

5. To remove the SBB, press its two mounting tabs together to release it from the shelf, and pull it out using both hands (see Figure 9-10).

Figure 9–10 SBB Warm Swap



While the OCP LEDs are flashing, any SBBs on the quiesced port that have status LEDs also flash.

> **NOTE**
>
> The length of time required for I/O to stop can vary from zero seconds to several minutes, depending on load, device type, and cache status.

After you remove the SBB, the flashing pattern on the OCP stops, and normal operation on the ports resumes. At this time the removed SBB's port LED turns on. The LED stays on until the SBB is returned to its slot or until another SBB is inserted in the slot. The remaining port LEDs turn off.

9.8.1.4 Device Replacement

Use a replacement device of the same type as the removed device. Otherwise, subsystem failures such as the establishment of storagesets may occur. Use the following procedure to replace a device:

1. Quiesce the SBBs port by pressing and holding the port button for the SBB. Continue holding the button until all amber OCP LEDs light.

> **NOTE**
>
> Only one port may be quiesced at any time.
>
> If the button is not held long enough, or multiple buttons are pushed in quick succession, all buttons are ignored (no ports are quiesced). You must press and hold the button again to quiesce the port.

2. Wait until the chosen port LED flashes alternately with the other port LEDs (this indicates I/O has stopped). The alternating pattern flashes for approximately 30 seconds, during which you may insert the SBB.

   If the pattern does not appear after a minute or two, another shelf is asserting a fault signal that prevents any quiesce function on this controller. To correct the problem, you must locate the suspect shelf and do one of three things:

   - Remove all devices from the shelf.
   - Disconnect the shelf's SCSI device cables (refer to Section 9.5).
   - Repair/replace the shelf power supply (refer to Section 9.7).

While the OCP LEDs are flashing, any SBBs on the quiesced port that have status LEDs also flash.

NOTE

The length of time required for I/O to stop can vary from zero seconds to several minutes, depending on load, device type, and cache status.

3. Hold the SBB in both hands, and firmly push it into the shelf until you hear the mounting tabs snap into place.

9.8.1.5 Restoring the Device to the Configuration

After you insert the SBB, the flashing pattern on the OCP stops, and normal operation on the ports resumes. At this time the port LEDs turn off.

- If you inserted a new device in a previously unused slot, that port's LED remains lit until the device is added by entering the following command (see Appendix B):

```
CLI>    ADD DISK container-name SCSI-location
```

- If the new disk is to be part of a storageset, you must delete the storageset from the configuration and create (ADD) it again.
- Initialize a newly inserted disk by entering the following:

```
CLI>    INITIALIZE container
```

  where *container* is either the disk, or a group of disks linked as a storageset.

  This initializes the metadata on each disk in the container, including the one that was just swapped.

CAUTION

If you think you have failed to perform warm swap exactly as stated here, you should reinitialize the controller. Otherwise, the controller may perform unpredictably.

Remember to close and lock cabinet doors after finishing the device warm swap.

9.8.2 Warm Swapping Controllers

Use warm swap to safely remove and replace one controller in a dual-redundant configuration. When you warm swap a controller, you are changing out a controller in the most transparent method available to the subsystem.

Performing warm swap involves removing one controller, while forcing the other controller into failover. Because the remaining controller executes failover, it assumes control of the absent controller's devices. This minimizes impact to system performance and downtime.

> **NOTE**
>
> You must warm swap only one controller at a time. Never attempt to remove both controllers in your dual-redundant configuration using warm swap.
>
> You must have a replacement controller available prior to starting warm swap. Otherwise, you must terminate the warm swap program and restart it later when you have a replacement.

9.8.2.1 Tools Required

You need the following tools to warm swap a controller:

- ESD strap
- 3/32-inch Allen wrench
- Flat-head screwdriver
- Small flat-head screwdriver

9.8.2.2 Precautions

Refer to Chapter 1 for ESD, grounding, module handling, and program card handling guidelines.

Ground yourself to an appropriate place on the cabinet before servicing the controller module.

9.8.2.3 Controller Removal

Use the following procedure to remove the controller.

1. If the controller you are removing has totally failed, proceed to step 2.

   If the controller you are removing has any I/O outstanding or devices on line, shut down that controller by using the following command:

   ```
   CLI> SHUTDOWN THIS_CONTROLLER
   ```

2. Connect either a virtual terminal connection or a maintenance terminal to the controller you are *not* removing.

3. Enter the RUN C_SWAP command. The system responds with the following:

```
Controller Warm Swap, Software Version -V2.0

*** Sequence to REMOVE other SWXRC-04 has begun. ***

Do you wish to REMOVE the other SWXRC-04 Y/N [N]?
```

4. Enter "Y" to continue the procedure.

```
Will its cache module also be removed Y/N [N]?
```

5. Enter "Y" only if you will be removing the controller's cache module as well.

```
Killing other controller.
Attempting to quiesce all ports.

Port 1 quiesced.
Port 2 quiesced.
Port 3 quiesced.
Port 4 quiesced.
Port 5 quiesced.
Port 6 quiesced.

All ports quiesced.

Remove the other SWXRC-04 (the one WITHOUT a blinking
green LED) within 2 minutes.
```

NOTE

Do not remove the controller with the blinking green LED reset (//) button.

6. You have 2 minutes to remove the controller following the steps described in Table 9-1. Your terminal will update you with the time remaining to complete the removal procedure, as shown in the following example:

```
Time remaining 1 minute, 40 seconds.
```

NOTE

If you fail to remove the controller within 2 minutes, the subsystem restarts the quiesced ports, and you must begin this procedure again.

Table 9–1 Module Removal

Step	Description
1	Ground yourself to an appropriate place on the cabinet.
2	Unlock and open the cabinet doors to gain access to the controllers.
3	Unsnap and remove the ESD shield covering the program card (refer to Figure 9-1).
4	Remove the program card by pushing the eject button (refer to Figure 9-1) next to the card. Pull the card out and save it for use in the replacement controller module.
5	With a small flat-head screwdriver, loosen the captive screws on the trilink connector and remove the trilink from the front of the controller. You must work around any SCSI cable or terminator connections when removing the trilink. Do not remove cables or terminators from the trilink or you will interrupt the host SCSI bus.
6	Loosen the four screws on each side of the front bezel.
7	Use a gentle up-and-down rocking motion to loosen the module from the shelf backplane.
8	Slide the module out of the shelf (noting which rails the module was seated in) and place on an approved ESD work surface or mat.
9	If necessary, you may now remove the cache module as described in Section 9.2.4.3 or 9.2.1.4.

Once you remove the controller, you see the following displays as the subsystem uses the remaining controller to service the quiesced ports:

```
Restarting ALL ports. Port 1 restarted.
Port 2 restarted.
Port 3 restarted.
Port 4 restarted.
Port 5 restarted.
Port 6 restarted.
```

9.8.2.4 Controller Replacement

Use the following procedure to replace the controller.

1. The system prompts you with the following to replace the controller:

   ```
   Do you have a replacement SWXRC-04 readily available
   [N]?
   ```

   Try to have a replacement available. If you do not have one, you must answer with "N." Then, the warm swap sequence will terminate, and you must restart the routine later when you have a replacement.

2. When you find a replacement, you can restart the sequence by entering the RUN C_SWAP command again. The system responds with the following:

   ```
   Do you have a replacement SWXRC-04 readily available
   [N]?
   ```

   Answer "Y" if you have the controller.

3. The following is displayed next:

```
*** Sequence to INSERT other SWXRC-04 has begun. ***
Do you wish to INSERT the other SWXRC-04 [N]?
```

Answer Y to insert the controller.

Remember to first reinsert the cache module if applicable.

```
Attempting to quiesce all ports.
   Port 1 quiesced.
   Port 2 quiesced.
   Port 3 quiesced.
   Port 4 quiesced.
   Port 5 quiesced.
   Port 6 quiesced.
All ports quiesced.
```

Insert the cache module, then insert the other SWXRC-04, WITHOUT its program card, and press Return

Insert the cache (if applicable) and controller now. Follow the steps outlined in Table 9-2.

Table 9–2 Module Replacement

Step	Description
1	Ground yourself to an appropriate place on the cabinet.
2	You should replace the cache module now, if you removed it. Refer to Section 9.2.4.4 or 9.2.1.5.
4	Slide the controller module into the shelf.
5	Use a gentle up-and-down rocking motion to help seat the module into the backplane. Press firmly on the module until it is seated. Finally, press firmly once more to make sure the module is seated.
6	Tighten the four screws on the front bezel.
7	Connect a maintenance terminal to the maintenance port of the other controller (the one you did not replace) if one is not already connected.

Once you insert the controller, you will see the following displayed as the subsystem restarts service on the quiesced ports:

```
Restarting ALL ports.
   Port 1 restarted.
   Port 2 restarted.
   Port 3 restarted.
   Port 4 restarted.
   Port 5 restarted.
   Port 6 restarted.
The configuration has two controllers.
```

5. Follow the steps in the system message:

   ```
   The Controller Warm Swap program has terminated. To
   restart the other controller:
   1) Enter the RESTART OTHER command.
   2) Press and hold the Reset button (//) while inserting
      the program card.
   3) Release Reset (//) and the controller will
      initialize.
   4) Configure new controller by referring to the
      StorageWorks Controller User's Guide.
   ```

   If the controller initializes correctly, its green reset LED begins to flash at 1 Hz. If an error occurs during initialization, the OCP displays a code. If necessary, refer to *Chapter 7* to analyze the code.

6. Snap the ESD shield into place over the program card.
7. Restore parameters for the new controller using the steps in Section 9.8.2.5.

9.8.2.5 Restoring Parameters

CAUTION

SET FAILOVER establishes controller-to-controller communication and copies configuration information. Always enter this command on one controller only. COPY=*configuration-source* specifies where the good configuration data are located. *Never* blindly specify SET FAILOVER. Know where your good configuration information resides before entering the command.

A new controller has no initial parameters. The parameters from the controller that was not removed will be copied over to the new controller when failover mode is reestablished. To do so, perform the following steps.

1. Enter the following command from the surviving controller (the controller you did *not* replace) to copy configuration information to the new controller:

   ```
   CLI>    SET NOFAILOVER
   CLI>    SET FAILOVER COPY=THIS_CONTROLLER
   ```

   SWXRC-04 controllers return to the dual-redundant configuration and restart after entering this command.

2. Connect the host port cable to the front of the controller. Do not connect the controllers in a dual- redundant pair to different host CPUs.

   Connect the SCSI cable trilink connector to the front of the controller and tighten its captive screws with a small flat-head screwdriver. You must work around any SCSI cable or terminator connections when replacing the trilink. Do *not* remove cables or terminators from the trilink or you will interrupt the host SCSI bus.

3. If you wish, you may disconnect the maintenance terminal. The terminal is not required for normal controller operation.
4. Close and lock the cabinet doors.

A

Field Replaceable Units

This appendix lists SWXRC-04 controller field replaceable units (FRUs), required tools and equipment, and related FRUs.

A.1 Controller Field Replaceable Units

The following FRUs come with the various controller modules. Part numbers are correct as of publication of this guide but are subject to change. Always verify your information in case part numbers or ordering methods have changed.

Table A–1 SWXRC-04 FRUs

FRU	Part Number
SWXRC-04 controller module (including bezel and trilink connector)	70-31457-02
32 MB cache module	54-22910-01
StorageWorks SWXRC-04 program flash card (blank)	BG-QHD30-1A
SCSI-2 device port cables	17-03566-15
Trilink connector (included in 70-31457-02)	12-39921-01
68-pin SCSI bus terminator	12-37004-03
Battery, sealed lead acid	12-40235-01

Table A–1 SWXRC-04 FRUs (continued)

FRU	Part Number
Bracket, dual battery	74-47465-02
Cover, memory card (ESD shield)	74-46416-01

A.2 Required Tools and Equipment

The following tools and equipment are required for controller maintenance:

- Portable antistatic kit, part number 29-26246-00
- ESD mat-for all module replacement service
- 5/32-inch Allen wrench-for opening the doors of an SWXSC-Dx-series data center cabinet.
- Flat-head screwdriver-for replacing host cables
- Small flat-head screwdriver-for replacing trilink connectors while SCSI host cables are attached
- An EIA-423 compatible terminal is needed for setting the initial configuration. When using this terminal, a connecting cable (between the terminal and the controller) that supports EIA-423 communication is required.

A.3 Related Field Replaceable Units

The following FRUs are related to the SWXRC-04 controllers. (Refer to the appropriate StorageWorks documentation for removal and replacement procedures for these components if not found in this guide.)

Table A-2 Controller Related FRUs

FRU	Option Number
Controller shelf (with backplane)	SWXSS-01
Storage Enclosure	SWXSC-AA
Device shelf (with backplane)	SWXSS-06
Shelf power supply	P-131 P-145 P-150
NULL modem DECconnect laptop 9-pin cable	H8571-J
DEC connect cable	BC16E-xx+

+ Where xx equals the length in feet

B

Command Line Interpreter

This appendix describes the use and command syntax of the SWXRC-04 controller CLI.

> **NOTE**
>
> The display outputs of some CLI commands may depict information fields not related to the SWXRC-04 controller or to devices supported by the controller. Inclusion of information on such device support by certain utilities does not imply support by the controller. The information contained in this chapter is included for reference only.

This appendix provides the following information:

- An overview of the Command Line Interpreter (CLI)
- A description of how to access and exit the CLI
- Device configuration examples using the CLI
- A comprehensive list of all CLI commands
- CLI error messages the operator may encounter

B.1 Command Line Interpreter Overview

The CLI is the user command line interface to the controller. The CLI enables you to add to or modify the controller's configuration using CLI commands. The following sections explain how to set up a controller, to define a storage subsystem configuration, and to modify it when needed.

B.2 Accessing the CLI

The CLI is implemented in the controller's firmware, and can be accessed in either of two ways:

- Using an EIA-423-compatible maintenance terminal-The SWXRC-04 controller provides an EIA-423 hardware port to which a maintenance terminal can be connected. You can access the CLI directly via the EIA-423 port.
- Using a virtual host terminal-You can use a virtual maintenance terminal program to access the CLI from the host. The details of connecting to the CLI with a virtual host terminal depend upon which operating system and terminal program you are using. The CLI commands and the way you use them remains unchanged.

B.2.1 Accessing the CLI Using a Maintenance Terminal

The initial controller configuration is normally set using a maintenance terminal connected to the EIA-423 terminal port on the controller. Thereafter, you can use a virtual host terminal to modify the controller's configuration.

> **NOTE**
>
> The CLI prompt has been factory-set to be "SWXRC>". See the information pertaining to the SET THIS_CONTROLLER AND SET OTHER_ CONTROLLER commands for details on changing the CLI prompt..

To access the CLI using a maintenance terminal, set the terminal for 9600 baud/8 data bits/1 stop bit/no parity, connect the terminal to the EIA-423 port, and make sure it displays the SWXCR> prompt. If the terminal display is not at the prompt, press the RETURN key to display it.

B.2.2 Setting the Initial Configuration for a Single Controller

After installing a single controller, use the CLI to define its initial parameters, as follows:

1. Access the CLI via either a maintenance terminal or virtual host terminal connection.
2. Enter the following command to set a valid controller ID:

```
SWXRC> SET THIS_CONTROLLER ID=n
```

Where n is the SCSI target ID(s) (0-7).

> **NOTE**
>
> Always restart the controller after setting the ID.

3. Restart the controller either by pressing the green reset (//) button, or by entering the following command:

```
SWXRC> RESTART THIS_CONTROLLER
```

4. After the controller restarts and returns to the CLI prompt, enter the following command to verify the controller's parameters:

```
SWXRC> SHOW THIS_CONTROLLER FULL
```

The controller is now ready for storage subsystem configuration.

B.2.3 Setting the Initial Configuration for Dual-Redundant Controllers

In a dual-redundant configuration, one terminal can set both controller configurations. After installing both controllers, use the CLI to define their initial parameters as follows:

1. Access the CLI on one of the controllers via either a maintenance terminal or virtual host terminal connection.

2 Enter the following command to set a valid controller ID:

```
SWXRC> SET THIS_CONTROLLER ID=n
```

Where n is the SCSI target ID(s) (0-7) for the first controller.

3. Then enter the following command:

   SWXRC> **SET OTHER_CONTROLLER ID=*n***

Where n is the SCSI target ID(s) (0-7) for the companion controller. All SCSI IDs assigned to the controller pair must be unique, and there can be no more than a total of four (4) IDs assigned to the pair.

> **NOTE**
>
> Always restart the controllers after setting the ID.

4. Restart both controllers either by pressing the green reset (//) buttons, or by entering the following commands:

```
SWXRC> RESTART OTHER_CONTROLLER
SWXRC> RESTART THIS_CONTROLLER
```

5. Enter the following commands to verify the preceding parameters were set.

```
SWXRC> SHOW THIS_CONTROLLER FULL
SWXRC> SHOW OTHER_CONTROLLER FULL
```

> **CAUTION**
>
> The SET FAILOVER command establishes controller-to-controller communication and copies configuration information. Always enter this command on one controller only. COPY=*configuration-source* specifies where the good configuration data are located. *Never* blindly specify SET FAILOVER. Know where your good configuration information resides before entering the command.

Once the initial parameters of both controllers are set, the storage subsystem can be configured. After configuring the controller's devices, the dual-redundant controllers are normally set to failover using the SET FAILOVER command.

B.2.4 Configuring Storage Devices

To automatically configure devices on the controller, use either the CONFIG or CFMENU utility described in Chapter 8.

For manual configuration, the following steps add devices, storagesets, and logical units. Use the CLI to complete these steps so that the host recognizes the storage device.

1. Add the physical devices by using the following command:

```
SWXRC> ADD device-type device-name scsi-location
```

   For example:

```
SWXRC> ADD DISK DISK100 1 0 0
```

   where:

   device-type is the type of device to be added. The device type used with the SWXRC-04 controller is DISK.

device-name is the name to refer to that device. The name is referenced when creating units or storagesets.

SCSI-location is the PTL for the device. When entering the PTL, at least one space must separate the port, target, and LUN.

2. Add the storagesets for the devices. Storagesets include stripesets and RAIDsets. See "*Device Configuration Examples*" for examples of adding storagesets.

> **CAUTION**
>
> The INITIALIZE command destroys all data on a **container**.

3. Enter the following command to initialize the containers (devices, storagesets, or both) prior to adding logical units to the configuration.

```
SWXRC> INITIALIZE container-name
```

where *container-name* is a device or storageset that becomes part of a unit.

When initializing a single-device container:

- If NOTRANSPORTABLE (the default) was specified when the device was added, a small amount of disk space was made inaccessible to the host and used for metadata. The metadata is now initialized.
- If TRANSPORTABLE was specified, any metadata on the device is now destroyed. See the information pertaining to the INITIALIZE command for details on metadata and when INITIALIZE is required.

4. Add the units that use either the devices or the storagesets built from the devices by entering the following command:

```
SWXRC> ADD UNIT logical-unit-number container-name
```

where:

logical-unit-number is the unit number the host uses to access the device.

container-name identifies the device or the storageset.

B.2.5 Device Configuration Examples

The following examples cover the majority of configurations and the method of defining those configurations.

Initial Single Controller Configuration of an SWXRC-04 Controller

```
SWXRC> SET THIS_CONTROLLER ID=0
SWXRC> RESTART THIS_CONTROLLER
[this controller restarts at this point]
```

Initial Dual-Redundant Controller Configuration of an SWXRC-04 Controller

```
SWXRC> SET THIS_CONTROLLER ID=(0,1)
SWXRC> SET THIS_CONTROLLER PREFERRED_ID=0
SWXRC> SET FAILOVER COPY=THIS_CONTROLLER
SWXRC> RESTART OTHER_CONTROLLER
```

[other controller restarts at this point]

```
SWXRC> RESTART THIS_CONTROLLER
```

[this controller restarts at this point]

Setting the Terminal Speed and Parity

```
SWXRC> SET THIS_CONTROLLER TERMINAL_SPEED=19200
NOTERMINAL_PARITY
```

Garbage will appear on the terminal after setting the controller's terminal speed until you set the terminal's speed to match the controller's new terminal speed.

Creating a Unit from a Disk Device

```
SWXRC> ADD DISK DISK0 2 0 0
SWXRC> INITIALIZE DISK0
SWXRC> ADD UNIT D0 DISK0
```

Creating a Unit from a Four-Member Stripeset

```
SWXRC> ADD DISK DISK0 1 0 0
SWXRC> ADD DISK DISK1 2 0 0
SWXRC> ADD DISK DISK2 3 0 0
SWXRC> ADD DISK DISK3 1 1 0
SWXRC> ADD STRIPESET STRIPE0 DISK0 DISK1 DISK2 DISK3
Warning 3000: This storageset is configured with more than one
disk per port. This causes a degradation in performance
SWXRC> INITIALIZE STRIPE0
SWXRC> ADD UNIT D0 STRIPE0
```

Creating a Unit from a Five-Member RAIDset

```
SWXRC> ADD DISK DISK0 1 0 0
SWXRC> ADD DISK DISK1 2 0 0
SWXRC> ADD DISK DISK2 3 0 0
SWXRC> ADD DISK DISK3 1 1 0
SWXRC> ADD DISK DISK4 2 1 0
SWXRC> ADD RAIDSET RAID9 DISK0 DISK1 DISK2 DISK3 DISK4
Warning 3000: This storageset is configured with more than one
disk per port. This causes a degradation in performance
SWXRC> INITIALIZE RAID9
SWXRC> ADD UNIT D0 RAID9
```

Creating a Unit from a Disk Device and Setting the Write Protection

```
SWXRC> ADD DISK DISK0 2 0 0
SWXRC> INITIALIZE DISK0
SWXRC> ADD UNIT D0 DISK0 WRITE_PROTECT
```

Setting the Write Protection for an Existing Unit

```
SWXRC> ADD DISK DISK0 2 0 0
SWXRC> INITIALIZE DISK0
SWXRC> ADD UNIT D0 DISK0
SWXRC> SET D0 WRITE_PROTECT
```

Renumbering Disk Unit 0 to Disk Unit 100

```
SWXRC> ADD DISK DISK0 2 0 0
SWXRC> INITIALIZE DISK0
SWXRC> ADD UNIT D0 DISK0
SWXRC> DELETE D0
SWXRC> ADD UNIT D100 DISK0
```

No INITIALIZE is required because DISK0 has already been initialized.

Creating a Transportable Unit from a Disk Device

```
SWXRC> ADD DISK DISK0 2 0 0 TRANSPORTABLE
SWXRC> INITIALIZE DISK0
SWXRC> ADD UNIT D0 DISK0
```

[or]

```
SWXRC> ADD DISK DISK0 2 0 0
SWXRC> SET DISK0 TRANSPORTABLE
SWXRC> INITIALIZE DISK0
SWXRC> ADD UNIT D0 DISK0
```

Changing the Replacement Policy of a RAIDset

```
SWXRC> ADD DISK DISK0 1 0 0
SWXRC> ADD DISK DISK1 2 0 0
SWXRC> ADD DISK DISK2 3 0 0
SWXRC> ADD DISK DISK3 4 0 0
SWXRC> ADD DISK DISK4 5 0 0
SWXRC> ADD RAIDSET RAID9 DISK0 DISK1 DISK2 DISK3 DISK4
SWXRC> INITIALIZE RAID9
SWXRC> ADD UNIT D0 RAID9
SWXRC> SET RAID9 POLICY=BEST_FIT
```

The replacement policy can be changed at any time.

Deleting the Unit, Stripeset, and All Disks Associated with a Stripeset

```
SWXRC> DELETE D0
SWXRC> DELETE STRIPE0
SWXRC> DELETE DISK0
SWXRC> DELETE DISK1
SWXRC> DELETE DISK2
SWXRC> DELETE DISK3
```

B.3 CLI Commands

The following subsections describe the valid commands in the CLI, along with their required parameters and qualifiers. Examples are given after the command format, parameters, description, and qualifiers.

ADD DISK

Adds a disk drive to the list of known disk drives.

Format

```
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, **T** designates the target ID of the device, (0–6, in a nonfailover configuration, or 0–5 if the controller is in a failover configuration), 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.

Description

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.

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:

1. `SWXRC>` **`ADD DISK RZ26_100 1 0 0`**

   Adds a non transportable disk to port 1, target 0, LUN 0 and names it RZ26_100.

2. `SWXRC>` **`ADD DISK DISK0 2 3 0 NOTRANSPORTABLE`**

   Adds a non transportable disk to port 2, target 3, LUN 0 and names it DISK0.

3. `SWXRC>` **`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.

Format

```
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.

Description

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.

Qualifiers

COPY=NORMAL (Default)
COPY=FAST

The COPY qualifier allows you to specify the speed at which mirrorset copies are performed. You may specify either NORMAL or FAST.

NORMAL (default) 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

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=LEAST_BUSY (Default)

READ_SOURCE=ROUND_ROBIN

READ_SOURCE=*device-container-name*

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. 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.

Example:

```
SWXRC> 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.

Format

```
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. (RAIDsetsize)

Description

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.

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. After finding the most closely matching devices, 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. After finding the best performing devices, 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)
RECONSTRUCT=FAST
NORECONSTRUCT

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 reconstructs the RAIDset at the fastest rate possible resulting in some loss of performance of the controller overall.

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:

1. `SWXRC>` **`ADD RAIDSET RAID9 DISK0 DISK1 DISK2 DISK3`**

   Creates a RAIDset with four disks (DISK0, DISK1, DISK2, and DISK3). The replacement policy is BEST_PERFORMANCE.

2. `SWXRC>` **`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.

3. `SWXRC>` **`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.

4. `SWXRC>` **`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.

Format

```
ADD SPARESET disk-container-name0
```

Parameters

disk-container-name

The disk drive container names to add to the spareset.

Description

The **SPARESET** is a pool of drives available to the controller to replace failing members of a RAIDset. The ADD SPARESET command adds a disk drive to the spareset and initializes the metadata on the drive so it can be used for replacement by a RAIDset.

Examples:

1. `SWXRC>` **`ADD SPARESET DISK0`**
   Adds one disk to the spareset.

ADD STRIPESET

Creates a stripeset from a number of containers.

Format

```
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.

Description

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.

Example

```
SWXRC> ADD STRIPESET STRIPE0 DISK0 DISK1 DISK2 DISK3
```

Creates a STRIPESET with four disks (DISK0, DISK1, DISK2, and DISK3).

ADD UNIT

Adds a logical unit to the controller.

Format

```
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=n1, n2, ...) command. You can not specify a target number that has not been previously specified by the SET THIS_CONTROLLER ID=n1, n2, ...) command.

container-name

The name of the container that is used to create the unit.

Description

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, mirrorsets, and RAIDsets 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.

Qualifiers for a Unit Created from a TRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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 MIRRORset

MAXIMUM_CACHED_TRANSFER_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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. 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 NOTRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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. 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 MIRRORset

MAXIMUM_CACHED_TRANSFER_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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. 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_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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.

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. 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_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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.

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Examples:

1. `SWXRC> ADD UNIT D0 DISK0`
   Creates disk unit number 0 from container DISK0.
2. `SWXRC> ADD UNIT D170 RAID9 WRITE_PROTECT`
   Creates disk unit number 170 from container RAID9 and write protects it.

CLEAR_ERRORS CLI

Stops displaying errors at the CLI prompt.

Format

```
CLEAR_ERRORS CLI
```

Description

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.

Example:

```
SWXRC>

All NVPM components initialized to their default settings.
SWXRC> CLEAR_ERRORS CLI
SWXRC>
```

Clears the message ("**All NVPM components initialized to their default settings.**") that was displayed at the CLI prompt.

CLEAR_ERRORS INVALID_CACHE

Clears all data from the cache and makes it usable by the specified controller.

Format

```
CLEAR_ERRORS INVALID_CACHE controller
```

Parameters

controller
Specifies which controller will clear the INVALID_CACHE condition. Either THIS_CONTROLLER or OTHER_CONTROLLER must be specified.

CAUTION

This command causes loss of customer data.

NOTE

Because this command causes loss of customer data, `INVALID_CACHE` must be completely spelled out, not abbreviated.

If a write-back cache module with unwritten cache data from another controller is installed on this controller, or if the write-back cache module with unwritten cache data is removed from this controller, an `INVALID_CACHE` error results. `CLEAR_ERRORS INVALID_CACHE` clears the invalid cache error, however *all customer data that was in cache is lost.*

For this reason, use great caution when considering using this command.

Example:

```
SWXRC> CLEAR_ERRORS INVALID_CACHE THIS_CONTROLLER
```

Clears all cache information from this controller's cache and clears the invalid cache error.

CLEAR_ERRORS LOST_DATA

Clears the lost data error on a unit.

Format

```
CLEAR_ERRORS LOST_DATA unit-number
```

Parameters

unit-number

Specifies the logical unit number that will have the lost data error cleared. The *unit-number* is the name given the unit when it was created using the `ADD UNIT` command.

> **CAUTION**
>
> This command causes loss of customer data. Because this command causes loss of customer data, `LOST_DATA` must be completely spelled out, not abbreviated. It may take up to 5 minutes to clear lost data. If customer data has been lost due to the removal or failure of the write-back cache, the lost data error is reported on the unit. `CLEAR_ERRORS LOST_DATA` clears the lost data error, however, (all customer data that had not been written to disk is lost). For this reason, use great caution when considering using this command.

Example:

```
SWXRC> CLEAR_ERRORS LOST_DATA D103
```

Clears the lost data error on disk unit D103.

CLEAR_ERRORS UNKNOWN

Clears the UNKNOWN error from a device.

Format

CLEAR_ERRORS UNKNOWN *device-name*

Parameters

device-name

Specifies the device name of the device with the UNKNOWN error.

> **NOTE**
>
> UNKNOWN must be completely spelled out, not abbreviated.

Description

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.

Example:

```
SWXRC> 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.

Format

```
CLEAR_ERRORS UNWRITEABLE_DATA unit-number
```

Parameters

unit-number

Specifies the logical unit number 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.

> **CAUTION**
>
> This command causes loss of customer data. Because this command causes loss of customer data, UNWRITEABLE_DATA must be completely spelled out, not abbreviated.

Description

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.

Example:

```
SWXRC> CLEAR_ERRORS UNWRITEABLE_DATA D103
```

Clears the unwriteable data error on disk unit D103.

DELETE *container-name*

Deletes a container from the list of known containers.

Format

```
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). 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.

Description

The spareset and failedset containers cannot be deleted. See DELETE SPARESET and DELETE FAILEDSET commands.

Examples:

1. `SWXRC> DELETE DISK0`

   Deletes DISK0 from the list of known containers.

2. `SWXRC> DELETE STRIPE0`

   Deletes STRIPE0 from the list of known containers.

3. `SWXRC> DELETE RAID9`

   Deletes RAID9 from the list of known containers.

DELETE FAILEDSET

Deletes a disk drive from the failedset.

Format

```
DELETE FAILEDSET disk-container-name0 [disk-container-nameN]
```

Parameters

disk-container-name0 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.

Description

The **FAILEDSET** is a group of drives that were removed from RAIDsets because they failed or were manually removed (via the SET *raidset-container-name* REMOVE=*disk-container-name* command). 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.

Examples:

1. `SWXRC> DELETE FAILEDSET DISK0`
   Deletes one disk from the failedset.
2. `SWXRC> DELETE FAILEDSET DISK0 DISK1 DISK2 DISK3 DISK4`
   Deletes five disks from the failedset.

DELETE SPARESET

Deletes a disk drive from the spareset.

Format

```
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.

Description

The **SPARESET** is a pool of drives available to the controller to replace failing members of a RAIDset. The DELETE SPARESET command removes disk drives from the spareset.

Examples:

1.
```
SWXRC> DELETE SPARESET DISK0
```
Deletes one disk from the spareset.

2.
```
SWXRC> DELETE SPARESET DISK0 DISK1 DISK2 DISK3 DISK4
```
Deletes five disks from the spareset.

DELETE *unit-number*

Deletes a unit from the list of known units.

Format

```
DELETE unit-number
```

Parameters

unit-number

Specifies the logical unit number that is to be deleted. The *unit-number* is the name given the unit when it was created using the ADD UNIT command.

Description

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.

Example:

```
SWXRC> DELETE D102
```

Deletes disk unit number 102 from the list of known units.

DIRECTORY

Lists the diagnostics and utilities available on THIS_CONTROLLER.

Format

```
DIRECTORY
```

Description

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.

For specific information about the diagnostics and utilities available, refer to the *StorageWorks Array Controllers HS Family of Array Controllers Service Manual*.

Example:

```
SWXCR> DIRECTORY
C_SWAPV25Z   D
CFMENUV25Z   D
CHVSN V25Z   D
CLCP  V25Z   D
CLONE V25Z   D
CONFIGV25Z   D
CRASH V25Z   D
DILX  V25Z   D
DIRECTV25Z   D
FLS   V25Z   D
FMU   V25Z   D

VTDPY V25Z   D
```

Displays directory listing.

EXIT

Exits the CLI and breaks the virtual terminal connection.

Format

```
EXIT
```

Description

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.

Examples:

1.
```
SWXRC> EXIT
Copyright Digital Equipment Corporation 1993, 1995. All rights
reserved. SWXRC-04 Firmware version V25Z-0, Hardware version A02
Last fail code: 20090010

Press " ?" at any time for help.

SWXRC>
```

HELP

Displays an overview for getting help.

Format

```
HELP
```

Description

The HELP command displays a brief description for using the question mark (?) to obtain help on any command or CLI function.

Examples:

1. SWXRC> **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 the user 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.

2. SWXRC> **SET ?**

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

Getting help on the SET command, using the (?) facility.

INITIALIZE

Initializes the metadata on the container specified.

Format

```
INITIALIZE container-name
```

Parameters

container-name

Specifies the container name to initialize.

Description

The INITIALIZE command initializes a container so a logical unit may be created from it. When initializing a single disk drive container, if NOTRANSPORTABLE was specified or allowed to default on the ADD DISK or SET *disk-container-name* commands, a small amount of disk space is made inaccessible to the host and used for metadata. The metadata is initialized. If TRANSPORTABLE was specified, 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.

It may take up to 2 minutes to initialize a RAIDset or stripeset.

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 or stripeset).

The INITIALIZE command specifically 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.

Qualifiers

CHUNKSIZE=*n*
CHUNKSIZE=DEFAULT (Default)

Specifies the chunksize to be used for RAIDsets and stripesets. The chunksize may be specified in blocks (CHUNKSIZE=*n*), or you can let the controller determine the optimal chunksize (CHUNKSIZE=DEFAULT). The CHUNKSIZE qualifier does not apply to mirrorsets

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.

Examples:

1. `SWXRC> INITIALIZE DISK0`

   Initializes container DISK0. If NOTRANSPORTABLE was specified (or allowed to default), metadata is written on the disk.

2. `SWXRC> INITIALIZE STRIPE0 CHUNKSIZE=20`

   Initializes container STRIPE0 and writes metadata on it.

3. `SWXRC> INITIALIZE RAID9 CHUNKSIZE=20`

   Initializes container RAID9 and writes metadata on it.

LOCATE

Locates units, storagesets, and devices by lighting the amber device fault LED on the front of the StorageWorks building block (SBB).

Format

```
LOCATE
```

Description

The LOCATE command illuminates the amber device fault LEDs (the lower LED on the front of an SBB) of the containers specified. The LOCATE command also can be used as a lamp test.

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, **T** designates the target ID of the device (0–6 in a nonfailover configuration or 0–5 if the controller is in a failover configuration), 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 LED corresponding with that device 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:

1. `SWXRC>` **`LOCATE DISK0`**
   Turns on the device fault LED on device DISK0.
2. `SWXRC>` **`LOCATE D102`**
   Turns on the device fault LEDs on all devices that make up disk unit number 102.
3. `SWXRC>` **`LOCATE DISKS`**
   Turns on the device fault LEDs on all configured disk devices.

MIRROR *disk-device-name1 container-name*

Allows you to convert a physical device to a one-member mirrorset.

Format

```
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 of letters A-Z, numbers 0-9, periods (.), dashes (-), or underscores (_), for a total of nine characters.

Description

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 firmware Version 2.5, the stripeset will then be incompatible with prior firmware versions.

Qualifiers

COPY=NORMAL (Default)
COPY=FAST

The COPY qualifier allows you to specify the speed at which mirrorset copies are preformed. 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.

Example:

```
SWXRC> MIRROR DISK210 MIRROR5
```

Creates a one-member mirrorset from DISK210 (a single disk).

REDUCE *disk-device-name1 [disk-device-nameN]*

Allows you to remove members from an existing mirrorset.

Format

```
REDUCE disk-device-name1 [disk-device-nameN]
```

Parameters

disk-device-name1

Specifies the name of the NORMAL mirrorset member to be removed.

[disk-device-nameN]

Specifies the name of the second mirrorset member to be removed.

Description

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.

> **NOTES**
>
> 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, of 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.
>
> 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.

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 autosparing 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.

Example:

```
SWXRC> REDUCE DISK210 DISK110
```

DISK210 and DISK110 are removed from their respective mirrorsets.

RENAME

Renames a container.

Format

```
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.

Gives a known container a new name by which to be referred.

Example:

```
SWXRC> RENAME DISK0 DISK100
```

Renames container DISK0 to DISK100.

RESTART OTHER_CONTROLLER

Restarts the other controller.

Format

```
RESTART OTHER_CONTROLLER
```

Description

The RESTART OTHER_CONTROLLER command flushes all user data from the other controller's write-back cache (if present), then restarts the other controller. If any user data cannot be flushed to disk, the controller does not restart unless the IGNORE_ERRORS qualifier is specified.

Specifying IMMEDIATE_SHUTDOWN causes the other controller to restart immediately without flushing any user data to the disks, even if drives are online to the host.

The RESTART OTHER_CONTROLLER command does not cause a failover to this controller in a dual-redundant configuration. The other controller restarts and resumes operations where it was interrupted.

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.

CAUTION

Customer data may be lost or corrupted if the IGNORE_ERRORS qualifier is specified.

IMMEDIATE_SHUTDOWN
NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE_SHUTDOWN is specified, the controller is immediately restarted without flushing user data from write-back cache to disk.

CAUTION

Customer data may be lost or corrupted if the IMMEDIATE_SHUTDOWN qualifier is specified.

Example:

1. `SWXRC>` **`RESTART OTHER_CONTROLLER`**

   Restarts the other controller.

RESTART THIS_CONTROLLER

Restarts this controller.

Format

```
RESTART THIS_CONTROLLER
```

Description

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_SHUTDOWN causes this controller to restart immediately without flushing any user data to the disks, even if drives are online to a host.

The RESTART THIS_CONTROLLER command does not cause a failover to the other controller in a dual-redundant configuration. This 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 this controller restarts.

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.

CAUTION

Customer data may be lost or corrupted if the IGNORE_ERRORS qualifier is specified.

IMMEDIATE_SHUTDOWN
NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE_SHUTDOWN is specified, the controller is immediately restarted without flushing user data from write-back cache to disk.

CAUTION

Customer data may be lost or corrupted if the IMMEDIATE_SHUTDOWN qualifier is specified.

Example:

1. `SWXRC>` **`RESTART THIS_CONTROLLER`**

   Restarts this controller.

RETRY_ERRORS UNWRITEABLE_DATA

Tries to write the unwriteable data on a unit.

Format

```
RETRY_ERRORS UNWRITEABLE_DATA unit-number
```

Parameters

unit-number

Specifies the logical unit number 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.

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 causing the unwriteable data should be corrected, and the write operation should then be attempted again. RETRY_ERRORS UNWRITEABLE_DATA attempts to write the unwriteable data error. No data is lost if the retry fails.

Example:

```
SWXRC> RETRY_ERRORS UNWRITEABLE_DATA D103
```

Attempts to write the cached data on disk unit D103 that was previously marked unwriteable.

RUN

Runs a diagnostic or utility on THIS_CONTROLLER.

Format

```
RUN program-name
```

Parameters

program-name

The name of the diagnostic or utility to be run. DILX is an example of the utilities and diagnostics that can be run from the CLI.

The RUN command enables various diagnostics and utilities on THIS_CONTROLLER. Diagnostics and utilities can be run (only) on the controller where the terminal is connected (see Chapter 8).

Example:

```
SWXRC> RUN DILX

Disk Inline Exerciser - version 2.0
                 .
                 .
                 .
```

Runs the DILX diagnostic.

SELFTEST OTHER_CONTROLLER

Runs a self-test on the other controller.

Format

```
SELFTEST OTHER_CONTROLLER
```

Description

The SELFTEST OTHER_CONTROLLER command flushes all user data from the other controller's write-back cache (if present), shuts down the other controller, then restarts it in DAEMON loop-on-self-test mode. The OCP reset (//) button must be pressed to take the other 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_SHUTDOWN causes the other controller to self-test immediately without flushing any user data to the disks.

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.

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.

CAUTION

Customer data may be lost or corrupted if the IGNORE_ERRORS qualifier is specified.

IMMEDIATE_SHUTDOWN
NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE_SHUTDOWN is specified, the controller will immediately start self-test without flushing user data from write cache to disk.

CAUTION

Customer data may be lost or corrupted if the IMMEDIATE_SHUTDOWN qualifier is specified.

Example:

```
SWXRC> SELFTEST OTHER_CONTROLLER
```

Starts the self-test on the other controller.

SELFTEST THIS_CONTROLLER

Runs a self-test on this controller.

Format

```
SELFTEST THIS_CONTROLLER
```

Description

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_SHUTDOWN causes this controller to self-test immediately without flushing any user data to the disks.

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.

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.

CAUTION

Customer data may be lost or corrupted if the IGNORE_ERRORS qualifier is specified.

IMMEDIATE_SHUTDOWN
NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE_SHUTDOWN is specified, the controller will immediately start self-test without flushing user data from write cache to disk.

CAUTION

Customer data may be lost or corrupted if the IMMEDIATE_SHUTDOWN qualifier is specified.

Example:

```
SWXRC> SELFTEST THIS_CONTROLLER
```

Starts the self-test on this controller.

SET *disk-container-name*

Changes the characteristics of a disk drive.

Format

```
SET disk-container-name
```

Parameters

disk-container-name

The name of the disk drive that will have its characteristics changed.

Changes the characteristics of a disk drive.

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.

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.

Example:

```
SWXRC> SET DISK130 TRANSPORTABLE
```

Sets DISK130 to transportable.

SET FAILOVER

Places THIS_CONTROLLER and OTHER_CONTROLLER into a dual-redundant configuration.

Format

```
SET FAILOVER COPY=configuration-source
```

Parameters

COPY=*configuration-source*

Specifies where the **good** copy of the device configuration resides.

If THIS_CONTROLLER is specified for *configuration-source*, all the device configuration information on THIS_CONTROLLER (the one that either the maintenance terminal is connected to or the virtual terminal is connected to) is copied to the other controller.

If OTHER_CONTROLLER is specified for *configuration-source*, all the device configuration information on the OTHER_CONTROLLER (the controller that either the maintenance terminal or the virtual terminal connection is *not* connected to) is copied to this controller.

The SET FAILOVER command places THIS_CONTROLLER and the OTHER_CONTROLLER in a dual-redundant configuration. After entering this command, if one of the two controllers fail, the devices and cache (if any) attached to the failed controller become available to and accessible through the operating controller.

CAUTION

All device configuration information on the controller *not* specified by the COPY= parameter is destroyed and overwritten by the configuration information found in the controller specified by the COPY= parameter.

BEFORE entering this command, make sure you know where your good configuration information is stored, or that you have a complete copy of the device configuration.

A considerable amount of work and effort is lost by overwriting a good configuration with incorrect information if the wrong controller is specified by the COPY= parameter.

Also note that due to the amount of information that must be passed between the two controllers, this command may take up to 1 minute to complete.

Example:

```
1. SWXRC> SET FAILOVER COPY=THIS_CONTROLLER
```

Places two controllers into a dual-redundant configuration, where the **good** data was on the controller that the maintenance terminal or virtual terminal connection was connected to.

```
2. SWXRC> SET FAILOVER COPY=OTHER_CONTROLLER
```

Places two controllers into a dual-redundant configuration, where the **good** data was on the controller that the maintenance terminal or virtual terminal connection was **not** connected to.

SET *mirrorset-container-name*

Changes the characteristics of a mirrorset.

Format

```
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.

Description

Changes the characteristics of a mirrorset.

Qualifiers

MEMBERSHIP=*number-of-members*

Allows you to increase or decrease the nominal membership of a mirrorset to the number specified by the *number-of-members.*

If the mirrorset membership is increased, and autosparing 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 autosparing 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 referenced 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-container-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, than 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 also may be marked as NORMALIZING if cache data is lost.)

Note that the nominal number of members in the mirrorset does not change. If autosparing 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-container-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=LEAST_BUSY (Default)
READ_SOURCE=ROUND_ROBIN
READ_SOURCE=*device-container-name*

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. 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=NORMAL (Default)
COPY= FAST

The SET *mirrorset-container-name* COPY 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:

1. `SWXRC>` **`SET MIRR9 POLICY=BEST_FIT`**

   Changes mirrorset MIRR9's policy to BEST_FIT.

2. `SWXRC>` **`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.

3. `SWXRC>` **`SET MIRR9 REPLACE=DISK320`**

   Adds disk DISK320 to the reduced mirrorset, MIRR9. A reconstruct operation begins immediately on DISK320.

SET NOFAILOVER

Removes THIS_CONTROLLER and OTHER_CONTROLLER (if reachable) from a dual-redundant configuration.

Format

```
SET NOFAILOVER
```

Description

The SET NOFAILOVER command removes THIS_CONTROLLER and the OTHER_CONTROLLER (if currently reachable) from a dual-redundant configuration. Before or immediately after entering this command, one controller should be physically removed because the sharing of devices is not supported by single controller configurations. The controller on which the command was entered is always removed from a dual-redundant state, even if the other controller is not currently reachable. No configuration information is lost when leaving a dual-redundant state.

Example:

```
SWXRC> SET NOFAILOVER
```

Removes the two controllers from a dual-redundant configuration.

SET OTHER_CONTROLLER

Changes the other controller's parameters (in a dual-redundant configuration the controller that the maintenance terminal is **not** connected to or the controller that is **not** the target of the DUP connection.

Format

```
SET OTHER_CONTROLLER
```

Description

The SET OTHER_CONTROLLER command allows you to modify the controller parameters of the other controller in a dual-redundant configuration.

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.

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.

Changing this policy on one controller in a dual-redundant controller configuration automatically changes it on the other controller. The parameter change takes effect immediately.

The CACHE_POLOCY setting affects RAIDsets and mirrorset 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 in made inoperative.
- If the CACHE_POLOCY 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.
- RAIDsets and mirrorset will not fail over.

  If the batteries go low after controller initialization, unwritten cache data 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.

> **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 should a power failure occur. Fully charged batteries will maintain the data for a minimum of 100 hours.

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.

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, and LUN 0, D5 would be target 0, LUN 5.

If two SWXRC-04 controllers are in a dual-redundant configuration, each controller will have the same IDs. When you change the IDs on one controller, the other will automatically be updated. A maximum of four IDs may be specified in any combination between the two controllers.

PREFERRED_ID=*n1[,nN]*
NOPREFERRED_ID

In a dual-redundant configuration, PREFERRED_ID defines which targets will be handled by the specified controller. If two or more PREFERRED_IDs are specified, they must be enclosed in parenthesis and separated by a comma. The only PREFERRED_IDs that can be specified must have already been configured using the ID= qualifier.

For example, if you configured a SWXRC-04 with IDs 0, 1, and 2, you could specify preferred ids 0, 1, and 2 in any combination on the two controllers. If one controller had preferred ids 0 and 1, it would handle unit numbers 0–7 and 100–107 and the other controller would handle unit numbers 200–207.

When you change the PREFERRED_IDs on one controller, the other controller will be automatically be updated to support the remaining (if any) IDs. By specifying NOPREFERRED_ID the controller does not respond to any target ID on the host's SCSI bus. However, in a dual-redundant mode, if the controller with PREFERRED_IDs specified were to fail, the controller with NOPREFERRED_ID would pick up the targets of the failed controller.

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 `swxrc>.`

TERMINAL_PARITY=ODD
TERMINAL_PARITY=EVEN
NOTERMINAL_PARITY (Default)

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. If the controller is in a dual-redundant configuration, the time is communicated to the other controller.

SET *RAIDset-container-name*

Changes the characteristics of a RAIDset.

Format

```
SET RAIDset-container-name
```

Parameters

RAIDset-container-name

The name of the RAIDset that will have its characteristics modified.

Changes the characteristics of a RAIDset.

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. After finding the most closely matching devices, 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. After finding the best performing devices, 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 RAID*set-container-name*, REPLACE=*disk-container-name*).

RECONSTRUCT=NORMAL (Default)
RECONSTRUCT=FAST
NORECONSTRUCT

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 reconstructs the RAIDset at the fastest rate possible resulting in some loss of performance of the controller overall.

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=).

REMOVE=*disk-container-name*

The disk removed via the REMOVE= command is added to the failedset.

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.

No other qualifiers to the SET *RAIDset-container-name* command may be specified if REPLACE is specified.

Examples:

1. `SWXRC>` **`SET RAID9 POLICY=BEST_FIT`**
   Changes RAIDset RAID9's policy to BEST_FIT.
2. `SWXRC>` **`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.
3. `SWXRC>` **`SET RAID9 REPLACE=SPAREDISK`**
   Adds disk SPAREDISK to the reduced RAIDset, RAID9. A reconstruct operation begins immediately on SPAREDISK.

SET THIS_CONTROLLER

Changes this controller's parameters (the controller that the maintenance terminal is connected to or the target of the DUP connection).

Format

```
SET THIS_CONTROLLER
```

The SET THIS_CONTROLLER command allows you to modify controller parameters on THIS_CONTROLLER in single and dual-redundant configurations.

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.

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.

Changing this policy on one controller in a dual-redundant controller configuration automatically changes it on the other controller. The parameter change takes effect immediately.

The CACHE_POLOCY setting affects RAIDsets and mirrorset 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 in made inoperative.
- If the CACHE_POLOCY 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.
- RAIDsets and mirrorset will not fail over.

  If the batteries go low after controller initialization, unwritten cache data 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.

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 should a power failure occur. Fully charged batteries will maintain the data for a minimum of 100 hours.

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.

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.

If two SWXRC-04 controllers are in a dual-redundant configuration, each controller will have the same IDs. When you change the IDs on one controller, the other will automatically be updated. A maximum of four IDs may be specified in any combination between the two controllers.

PREFERRED_ID=*n1[,nN]*
NOPREFERRED_ID

In a dual-redundant configuration, PREFERRED_ID defines which targets will be handled by the specified controller. If two or more PREFERRED_IDs are specified, they must be enclosed in parenthesis and separated by a comma. The only PREFERRED_IDs that can be specified must have already been configured using the ID= qualifier.

For example, if you configured a SWXRC-04 with IDs 0, 1, and 2, you could specify preferred ids 0, 1, and 2 in any combination on the two controllers. If one controller had preferred ids 0 and 1, it would handle unit numbers 0–7 and 100–107 and the other controller would handle unit numbers 200–207.

When you change the PREFERRED_IDs on one controller, the other controller will be automatically be updated to support the remaining (if any) IDs.

By specifying NOPREFERRED_ID the controller does not respond to any target ID on the host's SCSI bus. However, in a dual-redundant mode, if the controller with PREFERRED_IDs specified were to fail, the controller with NOPREFERRED_ID would pick up the targets of the failed controller.

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 `swxrc>.`

TERMINAL_PARITY=ODD
TERMINAL_PARITY=EVEN
NOTERMINAL_PARITY (Default)

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. If the controller is in a dual-redundant configuration, the time is communicated to the other controller.

Examples:

1. `SWXRC> SET THIS_CONTROLLER ID=5`

   Sets this SWXRC-04 controller so it responds to requests for target 5.

2. `SWXRC> SET THIS_CONTROLLER ID=(2,5)`

   Sets this SWXRC-04 controller so it responds to requests for targets 2 and 5.

SET *unit-number*

Changes the unit parameters.

Format

```
SET unit-number
```

Parameters

unit-number

Specifies the logical unit number to modify the software switches. The unit-number is the name given the unit when it was created using the ADD UNIT command.

The SET command is used to change logical unit parameters.

Qualifiers for a Unit Created from a TRANSPORTABLE Disk Drive

MAXIMUM_CACHED_TRANSFER_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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.

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 MIRRORset

MAXIMUM_CACHED_TRANSFER_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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.

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_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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.

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.

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_SIZE=*n*
MAXIMUM_CACHED_TRANSFER_SIZE=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.

It may take up to 5 minutes to flush unwritten data from the write-back cache once you disable write-back caching.

Example:

```
SWXRC> SET D1 WRITE_PROTECT NOREAD_CACHE
```

Sets the write protect and turns off the read cache on unit D1.

SHOW DEVICES

Shows physical devices and physical device information.

Format

```
SHOW DEVICES
```

The SHOW DEVICES command displays all the devices known to the controller. First disks are shown.

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each device.

Description

Information contained in the additional information is dependent on the device type.

Examples:

```
1. SWXRC> SHOW DEVICES

   Name        Type        Port Targ  Lun       Used by
   ----------------------------------------------------
   DISKI00     disk           1    0    0       D100
   DISK110     disk           1    1    0       D101
```

Shows a basic listing of devices attached to the controller.

```
2. SWXRC> SHOW DEVICES FULL

   Name        Type        Port Targ  Lun     Used by
   --------------------------------------------------
   DISK100     disk           1    0    0       D100
           DEC       RZ28   (C) DEC 442D
         Switches:
           NOTRANSPORTABLE
         Size: 4109470 blocks
   DISK110     disk           1    1    0       D101
           DEC       RZ29B  (C) DEC 0007
         Switches:
           NOTRANSPORTABLE
         Size: 8378028
```

Shows a full listing of devices attached to the controller.

SHOW DISKS

Shows all disk drives and drive information.

Format

```
SHOW DISKS
```

Description

The SHOW DISKS command displays all the disk drives known to the controller.

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each device.

Examples:

1. `SWXRC>` **`SHOW DISKS`**

```
Name        Type        Port Targ  Lun     Used by
--------------------------------------------------
DISK100     disk           1    0    0       D100
DISL110     disk           1    1    0       D101
```

Shows a basic listing of disks attached to the controller.

2. `SWXRC>` **`SHOW DISKS FULL`**

```
Name        Type        Port Targ  Lun     Used by
--------------------------------------------------
DISK100     disk           1    0    0       D100
        DEC       RZ28   (C) DEC 442D
      Switches:
        NOTRANSPORTABLE
      Size: 4109470 blocks
DISK110     disk           1    1    0       D101
        DEC       RZ29B  (C) DEC 0007
      Switches:
        NOTRANSPORTABLE
      Size: 8378028
```

Shows a full listing of disks attached to the controller.

SHOW *disk-container-name*

Shows information about a disk drive.

Format

```
SHOW disk-container-name
```

Parameters

disk-container-name

The name of the disk drive to be displayed.

The SHOW *disk-container-name* command is used to show specific information about a particular disk.

Example:

```
SWXRC> SHOW DISKI30

Name        Type        Port Targ  Lun     Used by
--------------------------------------------------
DISK130     disk           1    3    0       D103
        DEC      RZ28   (C) DEC 442D
      Switches:
        NOTRANSPORTABLE
      Size: 4109470 blocks
```

Shows a listing of disk 130

SHOW FAILEDSET

Shows the members of the failedset.

Format

```
SHOW FAILEDSET
```

Description

The SHOW FAILEDSET command displays all the disk drives that are members of the failedset.

Example:

```
SWXRC> SHOW FAILEDSET

Name        Storageset        Uses         Used by
--------------------------------------------------
FAILEDSET   failedset                      DISK220
                                           DISK320
```

Shows a listing of the members of the failedset.

SHOW MIRRORSETS

Shows all configured mirrorsets and any mirrorset-specific data related only to mirrorsets.

Format

```
SHOW MIRRORSETS
```

Description

The SHOW MIRRORSETS command displays all the mirrorsets known to the controller.

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each mirrorset.

Examples:

1.
```
SWXRC> SHOW MIRRORSETS

Name     Storageset                Uses      Used By
----------------------------------------------------
MIRR2    mirrorset                 DISK110   STR0
                                   DISK210
                                   DISK310
                                   DISK320

MIRR3    mirrorset                 DISK120   STR0
                                   DISK220

MIRR4    mirrorset                 DISK400   STR0
```

MIRR2 is a four-member mirrorset. MIRR3 is a two-member mirrorset. MIRR4 is a one-member mirrorset. All three mirrorsets are members of stripeset STR0.

2.

```
SWXRC> SHOW MIRRORSETS FULL

Name     Storageset                Uses       Used By
-------------------------------------------------------
MIRR2    mirrorset                 DISK110    STR0
                                   DISK210
                                   DISK310
                                   DISK320
    Switches:
      POLICY (for replacement) = BEST_PERFORMANCE
      COPY (priority) = NORMAL
                 READ_SOURCE = LEAST_BUSY
      MEMBERSHIP = 4, 4 members present
    State:
      DISK110    (member 0) is NORMAL
      DISK210    (member 1) is NORMAL
      DISK310    (member 2) is NORMAL
      DISK320    (member 3) 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:
      DISK120    (member 0) is NORMAL
      DISK120    (member 1) is NORMAL
    Size: 4109470 blocks

MIRR4    mirrorset                 DISK400    STR0
    Switches:
      POLICY (for replacement) = BEST_PERFORMANCE
      COPY (priority) = NORMAL
      READ_Source = LEAST_BUSY
      MEMBERSHIP = 1, 1 member present
    State:
      DISK330     (member 0) 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 mirrorset specified by *mirrorset-container-name*.

Format

```
SHOW mirrorset-container-name
```

Parameters

mirrorset-container-name

The name of the mirrorset to be displayed.

Description

The SHOW *mirrorset-container-name* command is used to show specific information about a particular mirrorset.

Example:

```
SWXRC> SHOW MIRR0

Name     Storageset     Uses           Used By
-----------------------------------------------------
MIRR0    mirrorset      DISK510        D205
      Switches:
       POLICY (for replacement)= BEST_PERFORMANCE
       COPY (priority) = NORMAL
       READ_Source = LEAST_BUSY
       MEMBERSHIP = 2, 1 member present
      State:
       DISK510 (member 0) is NORMAL
      Size: 4109470 blocks
```

Shows a complete listing of the mirrorset named MIRR0.

SHOW OTHER_CONTROLLER

Shows information for the other controller.

Format

```
SHOW OTHER_CONTROLLER
```

Description

Shows all controller, port, and terminal information for the other controller.

Qualifiers

FULL

If the FULL qualifier is specified, additional information is displayed after the basic controller information.

Example:

```
SWXRC> SHOW OTHER_CONTROLLER

Controller:
   SWXRC-04 ZG44400338 Firmware V25Z-0, Hardware A02
   Configured for dual-redundancy with ZG50100493
   In dual-redundant configuration
   SCSI address 6
   Time: NOT SET
Host port:
   SCSI target(s) (0, 1, 2, 3), Preferred target(s) (0,1)
Cache:
   32 megabyte write cache, version 2
   Cache is GOOD
   Battery is GOOD
   No unflushed data in cache
   CACHE_FLUSH_TIMER = 65535 seconds
   CACHE_POLICY = A
```

Shows the basic SWXRC-04 controller information.

```
SWXRC> SHOW OTHER_CONTROLLER FULL

Controller:
   SWXRC-04 ZG44400338 Firmware V25Z-0, Hardware  A02 Configured
   for dual-redundancy with ZG50100493
   In dual-redundant configuration
   SCSI address 6
   Time: NOT SET
   Host port:
   SCSI target(s) (0, 1, 2, 3), Preferred target(s) (0, 1)
Cache:
   32 megabyte write cache, version 2
   Cache is GOOD
   Battery is GOOD
   No unflushed data in cache
   CACHE_FLUSH_TIMER = 65535 (seconds)
   CACHE_POLICY = 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 9600 baud, eight bit, no parity, 1 stop bit
   Operation control: 00000004  Security state code: 65534
```

SHOW RAIDSETS

Shows RAIDsets and RAIDset information.

Format

```
SHOW RAIDSETS
```

Description

The SHOW RAIDSETS command displays all the RAIDsets known by the controller.

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each RAIDset.

Examples:

1.
```
SWXRC> SHOW RAIDSETS
Name        Storageset        Uses        Used by
------------------------------------------------
R0          raidset           DISK120     D206
                              DISK200
                              DISK320
                              DISK410

R1          raidset           DISK110
                              DISK220
                              DISK310
                              DISK400
```

Shows a basic listing of all RAIDsets.

2.

```
SWXRC> SHOW RAIDSETS FULL
Name      Storageset         Uses     Used by
-------------------------------------------------
R0        raidset            DISK120    D401
                             DISK200
                             DISK320
                             DISK410
     Switches:
      POLICY (for replacement) = BEST_PERFORMANCE
      RECONSTRUCT (priority) = NORMAL
      CHUNKSIZE = 63 blocks
     State:
      NORMAL
      DISK120  (member 0) is  NORMAL
      DISK200  (member 1) is  NORMAL
      DISK320  (member 2) is  NORMAL
      DISK410  (member 3) is  NORMAL
     Size: 12325392 blocks

R1          raidset           DISK110
                              DISK220
                              DISK310
                              DISK400

     Switches:
      POLICY (for replacement) = BEST_PERFORMANCE
      RECONSTRUCT (priority) = NORMAL
      CHUNKSIZE = 256 blocks
     State:
      UNKNOWN -- State only available when configured as a unit
     Size: 12325392 blocks
```

Shows a full listing of all RAIDsets.

SHOW *RAIDset-container-name*

Shows information about a RAIDset.

Format

```
SHOW RAIDset-container-name
```

Parameters

RAIDset-container-name

The name of the RAIDset to be displayed.

Description

The SHOW *RAIDset-container-name* command is used to show specific information about a particular RAIDset.

Example:

```
SWXRC> SHOW RAID9
Name     Storageset           Uses         Used by
-------------------------------------------------
RAID9    raidset              DISK110
                              DISK220
                              DISK310
                              DISK400
   Switches:
     POLICY (for replacement) = BEST_PERFORMANCE
     RECONSTRUCT (priority) = NORMAL
     CHUNKSIZE = 256 blocks
   State:
     UNKNOWN -- State only available when configured as a unit
   Size: 1235392 blocks
```

Shows a listing of RAIDset RAID9.

SHOW SPARESET

Shows the members of the spareset.

Format

```
SHOW SPARESET
```

Description

The SHOW SPARESET command displays all the disk drives that are members of the spareset.

Example:

```
SWXRC> SHOW SPARESET
Name          Storageset      Uses         Used by
-------------------------------------------------
SPARESET      spareset        DISK300
                              DISK510
                              DISK620
```

Shows a list of the members of the spareset.

SHOW STORAGESETS

Shows storagesets and storageset information.

Parameters

```
SHOW STORAGESETS
```

Description

The SHOW STORAGESETS command displays all the storagesets known by the controller. A storageset is any collection of containers, such as stripesets, mirrorsets, RAIDsets, the spareset, and the failedset.

Stripesets are displayed first, followed by mirrorsets, RAIDsets, the sparesets, and then the failedset containers.

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each storageset.

Examples:

1. `SWXRC>` **`SHOW STORAGESETS`**

```
Name      Storageset        Uses          Used by
----------------------------------------------
S1        stripeset        DISK300       D200
                           DISK310
                           DISK320
                           DISK330

M1         mirrorset       DISK230       D106
                           DISK410

R1         raidset         DISK100       D0
                           DISK110
                           DISK130
                           DISK210
SPARESET   spareset

FAILEDSET  faidedset
```

Shows a basic listing of all storagesets.

2. SWXRC> **SHOW STORAGESETS FULL**

```
Name          Storageset     Uses          Used by
-------------------------------------------------
S1            stripeset      DISK300      D200
                             DISK310
                             DISK320
                             DISK330
        Switches:
          CHUNKSIZE = 256 blocks
        State:
         DISK300   (member   0) is NORMAL
         DISK310   (member   1) is NORMAL
         DISK320   (member   2) is NORMAL
         DISK330   (member   3) is NORMAL
        Size:  16437880 blocks

M1            mirrorset     DISK230       D106
                            DISK410
        Switches:
          POLICY (for replacement) = BEST_FIT
          COPY (priority) = NORMAL
          READ_SOURCE = LEAST_BUSY
          MEMBERSHIP = 2, 2 members present
        State:
          DISK230   (member   0) is NORMAL
          DISK410   (member   1) is NORMAL
        Size: 4109470 blocks

R1           raidset        DISK100         D0
                            DISK110
                            DISK130
                            DISK210
        Switches:
          NOPOLICY (for replacement)
          RESCONSTRUCT (priority) = NORMAL CHUNKSIZE = 256
        State:
         DISK100   (member   0) is NORMAL
         DISK110   (member   1) is NORMAL
         DISK210   (member   2) is NORMAL
         DISK130   (member   3) is NORMAL
        Size:  12325292 blocks

SPARESET      spareset

FAILEDSET     failedset
```

Shows a full listing of all storagesets.

SHOW STRIPESETS

Shows stripesets and related stripeset information.

Format

```
SHOW STRIPESETS
```

Description

The SHOW STRIPESETS command displays all the stripesets known by the controller.

Qualifiers

FULL

If the FULL qualifier is specified, additional information may be displayed after each stripeset.

Examples:

1.
```
SWXRC> SHOW STRIPESETS
Name      Storageset      Uses         Used by
-------------------------------------------------
S0        stripeset       DISK300      D200
                          DISK310
                          DISK320
                          DISK330

S1        stripeset       M1      D100
                          M2
```

Shows a basic listing of all stripesets.

```
2. SWXRC> SHOW STRIPESETS FULL
   Name      Storageset      Uses     Used by
   ------------------------------------------------
   S1        stripeset       DISK300      D200
                             DISK310
                             DISK320
                             DISK330
          Switches:
            CHUNKSIZE = 256 blocks
          State:
            DISK300   (member   0) is NORMAL
            DISK310   (member   1) is NORMAL
            DISK320   (member   2) is NORMAL
            DISK330   (member   3) is NORMAL
          Size: 16437880 Blocks
   S2       stripeset        M1            D100
                             M2
          Switches:
            CHUNKSIZE = 256 blocks
          State:
            M1        (member   0) is NORMAL
            M2        (member   1) is NORMAL
          Size: 8218940 Blocks
```

Shows a full listing of all stripesets.

SHOW *stripeset-container-name*

Shows information about a specific stripeset.

Format

```
SHOW stripeset-container-name
```

Parameters

stripeset-container-name

The name of the stripeset to be displayed.

Description

The SHOW *stripeset-container-name* command is used to show specific information about a particular stripeset.

Example:

```
SWXRC> SHOW STRIPE0
Name      Storageset      Uses     Used by
--------------------------------------------------
STRIPE0   stripeset       DISK300      D200
                          DISK310
                          DISK320
                          DISK330
        Switches:
          CHUNKSIZE = 256 blocks
        State:
          DISK300   (member   0) is NORMAL
          DISK310   (member   1) is NORMAL
          DISK320   (member   2) is NORMAL
          DISK330   (member   3) is NORMAL
        Size: 16437880 Blocks
```

Shows a listing of stripeset STRIPE0.

SHOW THIS_CONTROLLER

Shows information for this controller.

Format

```
SHOW THIS_CONTROLLER
```

Description

Shows all controller, port, and terminal information for this controller.

Qualifiers

FULL

If the FULL qualifier is specified, additional information is displayed after the basic controller information.

Example:

```
SWXRC> SHOW THIS_CONTROLLER
```

```
Controller:
   SWXRC-04 ZG50100493 Firmware V25Z-0, Hardware A02
   Configured for dual-redundancy with ZG44400338 In dual-
   redundant configuration
   SCSI address 7
   Time: NOT SET
Host port:
   SCSI target(s) (0, 1, 2, 3), Preferred target(s) (2, 3)
Cache:
   32 megabyte write cache, version 2
   Cache is GOOD
   Battery is GOOD
   No unflushed data in cache
   CACHE_FLUSH_TIMER = 65535 (seconds)
   CACHE_POLICY = A
```

Shows the basic SWXRC-04 controller information.

```
SWXRC> SHOW THIS_CONTROLLER FULL
```

```
Controller:
   SWXRC-04 ZG50100493 Firmware V25Z-0, Hardware A02
   Configured for dual-redundancy with ZG44400338
   In dual-redundant configuration
   SCSI address 7
   Time: NOT SET
Host port:
        SCSI target(s) (0, 1, 2, 3), Preferred target(s) (2, 3)
```

```
Cache:
   32 megabyte write cache, version 2
   Cache is GOOD
   Battery is GOOD
   No unflushed data in cache
   CACHE_FLUSH_TIMER = 65535 (seconds)
   CACHE_POLICY = 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 9600 baud, eight bit, no parity, 1 stop bit
   Operation control: 00000004 Security state code: 88966
```

SHOW UNITS

Shows all units and unit information.

Format

```
SHOW UNITS
```

Description

The SHOW UNITS command displays all the units known by the controller. The command first lists disks.

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:

1. SWXRC> **SHOW UNITS**

```
LUN                                 Uses
------------------------------------------
  D0                                  R1
  D101                                S2
```

Shows a basic listing of units available on the controller.

2. SWXRC> **SHOW UNITS FULL**

```
LUN                            Uses
------------------------------------------
  D0                                  R0
         Switches:
           RUN      NOWRITE_PROTECT    READ_CACHE
           NOWRITEBACK_CACHE
           MAXIMUM_CACHED_TRANSFER_SIZE = 32
         State:
           ONLINE to the other controller
           Write cache good
           PREFERRED_PATH = OTHER_CONTROLLER
         Size:  8218940
  D101                                S2
         Switches:
           RUN      NOWRITE_PROTECT    READ_CACHE
           NOWRITEBACK_CACHE
           MAXIMUM_CACHED_TRANSFER_SIZE = 32
         State:
           ONLINE to the other controller
           PREFERRED_PATH = OTHER_CONTROLLER
         Size:  8218940
```

Shows a full listing of units available on the controller.

SHOW *unit-number*

Shows information about a specific unit.

Format

```
SHOW unit-number
```

Parameters

unit-number

The unit number of the unit that is to be displayed.

Description

The SHOW *unit-number* command is used to show specific information about a particular unit.

Example:

```
SWXRC> SHOW D206

  LUN                              Uses
---------------------------------------
  D206                             RAID9
       Switches:
          RUN         NOWRITE_PROTECT    READ_CACHE
          NOWRITEBACK_CACHE
          MAXIMUM_CACHED_TRANSFER_SIZE = 32
       State:
          ONLINE to the other controller
          Write cache good
          PREFERRED_PATH = OTHER_CONTROLLER
       Size: 12325392
```

Shows a listing of a specific disk unit.

SHUTDOWN OTHER_CONTROLLER

Shuts down and does not restart the other controller.

Format

```
SHUTDOWN OTHER_CONTROLLER
```

Description

The SHUTDOWN OTHER_CONTROLLER command flushes all user data from the other controller's write-back cache (if present), then shuts down the other 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_SHUTDOWN causes the other controller to shut down immediately without flushing any user data to the disks, even if drives are online to the host.

Qualifiers

IGNORE_ERRORS
NOIGNORE_ERRORS (Default)

If errors result when trying to write user data, the controller is not shut down unless IGNORE_ERRORS 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.

> **CAUTION**
>
> Customer data may be lost or corrupted if the IGNORE_ERRORS qualifier is specified.

IMMEDIATE_SHUTDOWN
NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE is specified, the controller shuts down immediately without flushing user data from write-back cache to disk.

> **CAUTION**
>
> Customer data may be lost or corrupted if the IMMEDIATE_SHUTDOWN qualifier is specified.

Example:

```
SWXRC> SHUTDOWN OTHER_CONTROLLER
```

Shuts down the other controller.

SHUTDOWN THIS_CONTROLLER

Shuts down and does not restart this controller.

Format

```
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_SHUTDOWN causes this controller to shut down immediately without flushing any user data to the disks.

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_ERRORS 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.

> **CAUTION**
>
> Customer data may be lost or corrupted if the IGNORE_ERRORS qualifier is specified.

IMMEDIATE_SHUTDOWN
NOIMMEDIATE_SHUTDOWN (Default)

If IMMEDIATE_SHUTDOWN is specified, the controller shuts down immediately without flushing user data from write-back cache to disk.

> **CAUTION**
>
> Customer data may be lost or corrupted if the IMMEDIATE_SHUTDOWN qualifier is specified.

Example:

```
SWXRC> SHUTDOWN THIS_CONTROLLER
```

Shuts down 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.

Format

```
UNMIRROR disk-device-name
```

Description

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).

Example:

```
SWXRC> UNMIRROR DISK130
```

Converts a DISK130 back to a single device. An error message prints if the disk is not a mirrorset.

B.4 CLI Messages

The following sections describe messages you can encounter during interactive use of the CLI.

B.4.1 Error Conventions

An Error (nnnn:) message means that the command did not complete. Except for a few of the failover messages (6000 series), no part of the command was executed. When encountering an error entering or exiting dual-redundant mode, some synchronization problems are unavoidable; in such a case, the error message tells you what to do to get things back in synchronization.

Multiple error messages may result from one command.

Items in angle brackets (<>) are replaced at run time with names, numbers, and so forth.

B.4.2 CLI Error Messages

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* specified is out of range.

Retry the ADD UNIT command with a correct number.

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.

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.

Error 1100: Disk unit numbers must start with the letter 'D'

Explanation: All disk unit numbers are of the form D*n*. This error is displayed when you add a disk unit if the letter “D” does not begin the unit number.

Retry the ADD command with the letter “D” at the beginning of the unit number.

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.

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.

Error 1130: The unit number cannot exceed <>

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 <>.

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.

Error 1150: A restart of THIS_CONTROLLER is required before units may be added

Explanation: You changed the target IDs that the 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.

Error 2000: Port must be 1 - <maximum port number>

Explanation: When adding a device, you specified a port less than 1 or greater than <>.

Retry the command specifying a port within the range given.

Error 2010: Target must be 0 - <maximum target number>

Explanation: When adding a device, you specified a target greater than <>.

In single controller configurations, <> is 6. In dual-redundant configurations, <> is 5.

Error 2020: LUN must be 0 - 7

Explanation: When adding a device, you specified a LUN greater than 7.

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.

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).

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).

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.

Error 3050: () 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.

Error 3060: () 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.

Error 3070: () 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.

Error 3080: () can't be deleted

Explanation: You attempted to delete the spareset or the failedset. These containers cannot be deleted.

Error 3090: () 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.

Error 3100: () 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.

Error 3110: () is not a member of (), cannot remove it

Explanation: When issuing a SET () REMOVE=(), the disk specified was not part of the container.

Check the device and container names and reissue the command.

Error 3120: () is already reduced. Another member cannot be removed

Explanation: When issuing a SET () REMOVE=(), the container was already in a reduced state. Add another disk before removing another member.

Error 3130: Unable to remove () from ()

Explanation: When issuing a SET () REMOVE=(), the controller was unable to remove the device from the RAIDset.

Check for error conditions, and if none exist, contact your service representative.

Error 3140: () is in a spareset. Remove it from the spareset first.

Explanation: When issuing a SET () REPLACE=(), the disk specified was part of the spareset. A disk to be used as a replacement must not be part of any configuration.

Error 3150: () is still part of a configuration. Delete upper configuration first.

Explanation: When issuing a SET () REPLACE=(), the disk specified was part of an existing configuration. A disk to be used as a replacement must not be part of any configuration.

Error 3160: () is not a disk. Can only use disks for replacement in a raidset.

Explanation: When issuing a SET () REPLACE=(), the device identified by () was not a disk.

Error 3170: () is not reduced. Cannot replace a member.

Explanation: When issuing a SET () REPLACE=(), the container specified was not reduced.

Remove a member before replacing it.

Error 3180: () has a replacement policy specified. Cannot manually replace a member.

Explanation: When issuing a SET () REPLACE=(), 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.

Error 3190: Unable to replace () in ()

Explanation: When issuing a SET () REPLACE=(), the controller was unable to replace the device into the RAIDset.

Check for error conditions, and if none exist, contact your service representative.

Error 3200: No other switches may be specified on a REMOVE operation.

Explanation: When issuing a SET () REMOVE=(), no other switches (such as POLICY) may be specified.

Error 3210: No other switches may be specified on a REPLACE operation.

Explanation: When issuing a SET () REPLACE=(), no other switches (such as POLICY) may be specified.

Error 3220: A REPLACE may not be done on a raidset that is not configured as a unit

Explanation: A REPLACE operation may not be done on a RAIDset that has not been configured as a unit.

Error 3230: () is reconstructing (). Only () may be removed

Explanation: When issuing a SET () REMOVE=() on a RAIDset that is already reconstructing, only the disk drive that is being reconstructed may be removed.

Error 3240: () can't be initialized

Explanation: Sparesets and failedsets cannot be initialized.

Check the name of the container that you wish to initialize and try again.

Error 3250: A REMOVE may not be done on a raidset that is not configured as a unit

Explanation: A RAIDset must be configured as a unit before a disk can be removed to reduce the RAIDset.

Create a unit from the RAIDset and then remove the member.

Error 3260: () is a TRANSPORTABLE disk. TRANSPORTABLE disks cannot be used by storagesets. Do a SET () 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.

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.

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).

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.

Error 4090: Module has invalid serial number. This controller cannot be used. Call your service representative.

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 your service representative.

Error 4100: Unable to RESTART other controller.

Explanation: A communication error occurred when trying to restart the other controller.

Retry the RESTART command.

Error 4110: Unable to SHUTDOWN other controller.

Explanation: A communication error occurred when trying to shutdown the other controller.

Retry the SHUTDOWN command.

Error 4120: Unable to SELFTEST other controller.

Explanation: A communication error occurred when trying to self-test the other controller.

Retry the SELFTEST command.

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.

Error 4140: Unable to lock the other controller's NV memory.

Explanation: Most configuration commands such as ADD, DELETE, and SET require both controllers in a dual-redundant configuration to be running so configuration changes can be recorded in both controllers. If one controller is not running, the above message results when you attempt to change the configuration.

Restart the other controller and try the command again, or SET NOFAILOVER on the remaining controller.

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 either by online units or errors when trying to rundown the units. Either rectify the problems on the problem units or issue the SHUTDOWN, RESTART or SELFTEST command with the OVERRIDE_ONLINE or IGNORE_ERRORS qualifiers.

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.

Either rectify the problems on the problem units or issue the SHUTDOWN, RESTART or SELFTEST command with the qualifier OVERRIDE_ONLINE or IGNORE_ERRORS.

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.

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> to <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.

Error 4190: The time must be specified in the format dd-mmm-yyyy:hh:mm:ss

Explanation: On the SWXRC-04 controllers only, the time must be specified as shown.

Retry the command using the correct time format.

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.

Error 4210: IDs specified as preferred must be a subset of the IDs specified by the ID= argument first

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.

Error 5000: A program name must be from 1 to 6 characters in length

Explanation: This error results from a “RUN <program name>”.

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.

Error 5020: The requested program is unknown.

Explanation: This error results from a “RUN <program name>“.

Explanation: Enter DIR to get a list of available programs.

Error 5030: Insufficient memory for request.

Explanation: This error results from a “RUN <program name>” resource problem. Retry the command later.

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 your service representative.

Error 6010: Other controller not present

Explanation: When asked to communicate with another controller (the result of any one of a number of commands), the other controller was found not to be running.

If the other controller is in the process of restarting, retry the command later. If the other controller is shut down or turned off, start it. If the other controller is no longer present, enter a SET NOFAILOVER command to take it out of dual-redundant mode.

Error 6020: Initial failover handshake not yet complete

Explanation: For a short period of time after start up, the two controllers must communicate to set up a dual-redundant mode. This setup time is typically less than 1 minute. If commands that require controller-to-controller communication are entered during this setup time, error 6020 results.

Retry the command later.

Error 6030: Unable to communicate with the other controller to setup FAILOVER

Explanation: Could not setup FAILOVER due to communication problems between the controllers.

Retry the command later.

Error 6040: The write of the other controller's configuration information did not succeed; information may be in an inconsistent state. Before further use both controllers should be removed from dual-redundant mode (SET NOFAILOVER) and then placed back into dual-redundant mode (SET FAILOVER) to assure consistency

Explanation: Communication was lost in the middle of a SET FAILOVER command.

Follow the instructions included in the error message.

Error 6050: Communication failure with other controller while putting controllers into dual-redundant mode. Reissue SET FAILOVER command

Explanation: Communication was lost in the middle of a SET FAILOVER command.

Follow the instructions included in the error message.

Error 6070: Illegal command – this controller not configured for dual-redundancy

Explanation: A command was entered to a single controller configuration that requires two controllers to be in dual-redundant mode.

If two controllers are supposed to be in dual-redundant mode, enter a SET FAILOVER command. If not, do not enter the command that resulted in the error.

Error 6080: Illegal command – this controller not currently in dual-redundant mode

Explanation: A command was entered to a dual-redundant-configured controller, but the other controller was not available for communication.

Restart the other controller and wait until it is communicating with this controller. If this controller is no longer supposed to be in dual-redundant mode, enter a SET NOFAILOVER command.

Error 6090: In failover no device may be configured at target 6 <device name> is at PTL <port><target><lun>

Explanation: Target addresses 6 and 7 are used by the controllers when in a dual-redundant configuration. When in a single controller configuration, target 6 is available for use by devices. If devices are configured at target 6 and you attempt to install a dual-redundant configuration, this error is displayed for all devices that use target 6 and the controllers will not be placed in a dual-redundant configuration.

Reconfigure the drives both logically and physically so that target 6 is not used.

Error 6110: Controllers already configured for failover

Explanation: A SET FAILOVER cannot be issued on a controller already in failover.

Error 6130: RAID5 in use on this controller but not enabled on the other controller.

Explanation: When trying to SET FAILOVER, it was discovered that there were RAID5 configurations on this controller but the other controller did not have the RAID5 feature enabled.

If RAID5 is licensed on the other controller, enable it. If it is not licensed, either contact your service representative for licensing information, or do not use the two controllers in dual-redundant mode, or do not use a RAID5 configuration.

Error 6140: Writeback cache in use on this controller but not enabled on the other controller.

Explanation: When trying to SET FAILOVER, it was discovered that there were write-back cache switches set on this controller but the other controller did not have the write-back cache feature enabled.

If write-back cache is licensed on the other controller, enable it. If it is not licensed, either contact your service representative for licensing information, or do not use the two controllers in dual-redundant mode, or do not use the write-back cache switches.

Error 6150: RAID5 in use on other controller but not enabled on this controller

Explanation: When trying to SET FAILOVER, it was discovered that there were RAID5 configurations on the other controller but this controller did not have the RAID5 feature enabled.

If RAID5 is licensed on this controller, enable it. If it is not licensed, contact your service representative for licensing information or do not use the two controllers in dual-redundant mode, or do not use a RAID5 configuration.

Error 6160: Writeback cache in use on other controller but not enabled on this controller.

Explanation: When trying to SET FAILOVER, it was discovered that there were write-back cache switches set on the other controller but this controller did not have the write-back cache feature enabled.

If write-back cache is licensed on this controller, enable it. If it is not licensed, either contact your service representative for licensing information or do not use the two controllers in dual-redundant mode, or do not use the write-back cache switches.

Error 6170: An <controller type> and <controller type> cannot configured for failover

Explanation: Two different controllers (such as the SWXRC-04 and the HSZ40) cannot be configured for failover.

Replace the other controller with the same model as this one and reissue the command.

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.

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.

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.

Error 7030: Unable to force write of unwriteable data

Explanation: A RETRY UNWRITEABLE_DATA command could not write the UNWRITEABLE_DATA.

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 your service representative.

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.

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.

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.

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.

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.

Error 9040: <name> conflicts with keyword <keyword>.

Explanation: The name given in an ADD command conflicts with a CLI keyword.

Specify another name.

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.

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.

Error 9080: <name> is already used

Explanation: An ADD command specified a name that is already in use.

Specify another name.

Error 9090: A <device type> cannot be used in a <storageset type>.

Explanation: The device specified cannot be used in the storageset specified.

Reexamine the configuration and correct the incompatibility.

Error 9100: A <storageset type> must have from <least> to <most> 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.

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 may no longer be initialized. The container that makes up the unit must be initialized before a unit is created out of the container.

Error 9170: ( ) at PTL ( ) 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.

Error 9180: ( ) at PTL ( ) Incorrect device type installed

Explanation: When a unit is added or initialized, the configuration of the devices that make up the unit is checked. If a non disk 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.

Error 9200: <name> conflicts with unit names

Explanation: This error results from an ADD command. Names in the format of D*n* when *n* is a number from 0 through 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.

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.

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.

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.

Ensure that DILX is not being run against the unit that is to be deleted, and retry the command.

Error 9250: Initialize of disk failed

Explanation: Unable to write metadata on disk.

Make sure the disk is not broken.

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.

Error 9270: No metadata found on container, unit not created. An INITIALIZE <container name> must be issued before this container may be used

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.

Error 9290: Communication failure with other controller, cannot check other controller's licensing

Explanation: Unable to communicate with the other controller to check licensing before creating a RAIDset or enabling write-back cache.

Check to make sure that both controllers are running. If one is broken, take this controller out of failover (SET NOFAILOVER) and reissue the command.

Error 9320: Bad write-back cache or battery on <_________> controller

Explanation: If you attempt to set write-back cache on a device or create a RAIDset, and there is a bad cache or cache battery on a controller, write-back is not set, or the RAIDset is not created and the above message is displayed.

Error 9330: NV memory write collision. Please try again

Explanation: Two people were trying to configure the CLI at the same time.

Check the configuration you were trying to modify to make sure it's unchanged and retry the command.

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.

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 your service representative.

Error 9400: Cannot rundown or allocate unit in order to delete it

Explanation: Retry the command. If this error persists, call your service representative.

Error 9410: Cannot delete unit – (_______) error exists on unit that must be cleared first. To clear error type: (_______)

Explanation: Units cannot be deleted if cache errors exist. Any cache errors must be cleared before a unit can be deleted.

Issue the <device type> command and then delete the unit.

Error 9420: Unit <device type> has unflushed data or a cache error and must be deleted on this controller

Explanation: When trying to set failover a unit with unflushed data or a cache error was detected on this controller.

Delete the unit as requested and then retry the SET FAILOVER command.

Error 9430: Cannot check if drives have unflushed data or cache errors on the other controller

Explanation: Communication error when trying to SET FAILOVER.

Retry the command. If this error persists, call your service representative.

Error 9440: Unit <device type> has unflushed data or a cache error and must be deleted on the other controller

Explanation: When trying to set failover a unit with unflushed data or a cache error was detected on the other controller.

Delete the unit as requested and then retry the SET FAILOVER command.

B.4.3 Warning 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.

B.4.4 CLI Warning Messages

Warning 1000: It is recommended that you read the controller product documentation to understand the significance of enabling WRITEBACK_CACHE, particularily 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.

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.

Warning 3010: Unable to check all device types that make up this storageset. If the storageset is made up of different device types, it may result in a storageset of reduced size

Explanation: This warning results from an ADD *storageset-type* command. Device types being added to a storageset are checked to assure that they are the correct device types. If one or more devices could not be checked, the above warning is displayed.

You should check all the devices to assure that they are correctly installed and configured.

Warning 3020: This storageset is configured with different device types. This may result in a storageset of reduced size

Explanation: This warning results from an ADD *storageset-type* command. Device types being added to a storageset are checked to assure that they are the same types. If all devices are not the same, the above warning is reported. Storageset size is determined by the size of the smallest device, so the storageset configured is 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, all devices that make up the storageset should be identical.

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.

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.

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.

Warning 6000: Communication failure with the other controller while taking controllers out of dual-redundant mode. Issue a SET NOFAILOVER command on the other controller

Explanation: This warning results from a SET NOFAILOVER command. This controller was unable to communicate with the other controller to notify it that it is no longer in dual-redundant mode. Typically, this occurs when the other controller has already been removed prior to the SET NOFAILOVER command.

Enter a SET NOFAILOVER command on the other controller as soon as possible.

Warning 6010: Licensing different between the two controllers

Explanation: If the licensing is not identical on both controllers in a dual-redundant configuration, the above warning is displayed.

You should check the licensing on both controllers and make sure they are identical.

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.

This means that no customer data was lost, so this warning is actually good.

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 your service representative.

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 your service representative.

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.

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.

Warning 9050: ( ) at PTL ( ) 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.

Warning 9060: ( ) at PTL ( ) 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.

Warning 9080: <device type> support is not licensed on <device type> 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 your service representative if you wish to purchase it.

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.

Warning 9100: Bad or low battery or bad write cache on <device type> 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.

Warning 9110: Bad or low battery or bad write cache on <device type> this unit cannot be used by <device type>

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.

Storage Subsystem Glossary

This chapter provides a glossary of Storage Subsystem terms.

ac distribution

The method of distributing ac power in a cabinet.

ac power supply

A power supply designed to produce dc power from an ac input.

adapter

Also called a *signal converter*. An adapter converts the protocol and hardware interface of one bus type into that of another without changing the function of the bus.

allocation class

A numerical value assigned to an integrated storage element to indicate which hosts on a cluster it is served by.

American National Standards Institute

See ANSI.

ANSI

American National Standards Institute. ANSI is an organization that develops and publishes electronic and mechanical standards.

array controller

A hardware/software device that facilitates communications between a host and one or more devices organized in an array. SWXRC-04 is an array controller.

bad block

A block containing a defect that:

- Exceeds the correction capability of the subsystem error correction scheme.
- Exceeds a drive-specified error threshold. Once a block exceeds this threshold, data integrity is not guaranteed.
- Imposes too great a strain on system performance. In this case, the subsystem still assures data integrity, but the extensive error correction required for each block access causes too great a strain on system performance.

bad block replacement

See *BBR*.

battery backup unit

See *BBU*.

BBR

Bad block replacement. BBR is the substitution of defect- free disk blocks for those found to have defects. BBR locates a replacement block, marks the bad block as replaced, and moves the data from the bad block to the replacement block.

BBU

Battery backup unit. A BBU extends power availability after the loss of primary ac power or a power supply to protect against the corruption or loss of data.

BIST

Built-in self-test. BIST is the internal self-test routine for the SWXRC-04-series controller module microprocessor chip.

block

A stream of data transferred as a unit. Block is used interchangeably with the term "sector" for disk drives to represent 512 bytes (for 16- and 32-bit host architectures) or 576 bytes (for 36-bit architectures). A block is the smallest data unit addressable on a subunit. It occupies a specific physical position relative to the index and is available for reading or writing once per disk rotation. The five types of blocks follow:

1. Diagnostic block-Used for drive read or write diagnostics. The diagnostic block area is not visible to the host operating system. However, it is visible to the controller. Diagnostic block addresses are 28 bits wide and are called diagnostic block numbers (DBNs).
2. External block-Contains the format control tables. The external block area is not visible to the host operating system. However, it is visible to the controller. External block addresses are 28 bits wide and are called external block numbers (XBNs).
3. Logical block-Contains the host applications area and the Replacement Control Table. All logical blocks are visible to the host operating system. Logical block addresses are 28 bits wide and are called logical block numbers (LBNs).
4. Physical block-Contains all the blocks on a subunit. DBNs, LBNs, RBNs, and XBNs are subsets of the physical block area. Physical block addresses are 28 bits wide and are called physical block numbers (PBNs).
5. Replacement block-A reserved block used as a replacement for a bad block on a subunit. Replacement block addresses are 28 bits wide and are called replacement block numbers (RBNs).

blower

An airflow device mounted in a StorageWorks shelf.

built-in self-test

See *BIST*.

cable distribution unit

See *CDU*.

carrier

Sometimes called an SBB carrier. A carrier is a standard, StorageWorks shelf-compatible, plastic shell into which a device can be installed.

CDU

Cable distribution unit. The CDU is the power entry device for StorageWorks cabinets. The unit provides the connections necessary to distribute ac power to cabinet shelves and fans.

CER

CLI error report. The CER spontaneously posts controller error messages to a terminal, if one is connected.

CLI

Command line interpreter. Operator command line interface for the SWXRC-04-series controller.

CLI error report

See *CER*.

cluster

A collection of processors, called nodes, attached to each other by a high-speed bus. These processors are independent and survivable. They may be general-purpose computers or special-purpose servers, providing a special set of services to the rest of the nodes.

cold swap

A method of device replacement that requires that power be removed from all shelves in a cabinet. This method is used when conditions preclude the use of the warm swap or hot swap methods.

command line interpreter

See *CLI*.

container

Data storage consisting of either a single disk device, or group of disk devices linked as a storageset. Examples of containers include disk drives, stripesets, mirrorsets, and RAIDsets.

controller

A hardware/software device that facilitates communications between a host and one or more devices. A controller translates bus protocols and hardware interfaces and adds functions to the host/device communication.

controller shelf

A StorageWorks shelf designed to contain controller and cache memory modules.

CRC

A checkword (polynomial checksum) generally appended to a disk data transfer. CRC is computed using data message bits as coefficients divided by a generating polynomial. The resulting remainder is the CRC. When a transmitter computes and transmits a CRC following a data transfer, the receiver can recompute and compare it with the received version to verify correct reception. EDC and ECC (both used by disks) are examples of CRC checkwords.

cyclic redundancy check

See *CRC*.

DAEMON

Diagnostic and execution monitor. DAEMON is a part of the SWXRC-04 controller's self-testing that includes port and cache initialization and self-test routines.

data center cabinet

A generic reference to the large cabinets, such as the SWXSC-Dx series, in which StorageWorks components can be mounted.

device driver

An operating system software module used to physically control an I/O device.

device shelf

A StorageWorks shelf designed to contain SBBs.

diagnostic and execution monitor

See *DAEMON*.

differential SCSI bus

A SCSI bus in which a signal's level is determined by the potential difference between two lines. A differential bus is more robust and less subject to electrical noise than is a single-ended bus.

DIGITAL Standard Disk Format

See *DSDF*.

DILX

Disk inline exerciser. DILX is diagnostic firmware used to test the data transfer capabilities of disk drives in a way that simulates a high level of user activity.

Disk Inline Exerciser

See *DILX*.

dual cabinet power configuration

A cabinet ac power configuration in which two ac sources and two ac power supplies are used to provide redundant dc power to each of the cabinet's SBB shelves.

dual data link

See *DDL*.

dual shelf power configuration

A cabinet ac power configuration in which one ac source and two ac power supplies are used to provide redundant dc power to each of the cabinet's SBB shelves.

dual-redundant

Two controllers in one controller shelf, sharing access to each other's devices. This configuration provides for one controller to take over the work of a failed controller.

ECC

Error correction code. The ECC is one or more cyclic redundancy check (CRC) words that enable detection of a mismatch between transmitted and received data in a communications system, or between stored and retrieved data in a storage system. The ECC allows for location and correction of an error in the received/retrieved data. All ECCs have limited correction power.

EDC

Error detection code. The EDC is one or more checksum words that enable detection of a mismatch between transmitted and received data in a communications system, or between stored and retrieved data in a storage system. The EDC has no data correction capability.

EIP

Error information packet (or event information packet). The EIP includes bytes of data translated into information explaining error events.

electromagnetic interference

See *EMI*.

electrostatic discharge

See *ESD*.

EMI

Electromagnetic interference. EMI is the impairment of a signal by an electromagnetic disturbance.

EMU

Environmental monitor unit. The EMU provides increased protection against catastrophic subsystem faults in the SWXSC-AA cabinet .

environmental monitor unit

See *EMU*.

error correction code

See *ECC*.

error detection code

See *EDC*.

error information packet

See *EIP*.

ESD

Electrostatic discharge. ESD is the discharge of a potentially harmful static electric voltage as a result of improper grounding.

event logging

See *EVL*.

EVL

Event logging. EVL is the spontaneous posting of EIP information to a connected terminal. The EVL display enables the user to read event log information without invoking a separate utility for translation.

EXEC

Firmware executive. EXEC is the kernel portion of the SWXRC-04 controller firmware.

extended status

An additional set of status information maintained by the drive that is of interest to a host error log. Extended status is drive-type specific and is not utilized by the controller except as input to the host error log and diagnostic processes.

failedset

A disk drive or group of disk drives that has been removed from a RAIDset due to a failure or manual removal. Disk drives in the failedset should be considered defective and should be tested, then repaired or replaced.

failover

A process that takes place when one controller fails in a dual-redundant configuration, and the other controller takes over service to the devices of the failed controller.

fan

An airflow device mounted in a StorageWorks cabinet.

fast, differential SCSI

See *FD SCSI*.

fast, wide, differential SCSI

See *FWD SCSI*.

Fault Management Utility

See *FMU*.

FD SCSI

Fast, differential SCSI. This differential SCSI bus has an 8-bit data transfer rate of 10 MB/s. See also FWD SCSI and SCSI.

field replaceable unit

See *FRU*.

filler panel

A sheet metal or plastic panel used to cover unused mounting areas in StorageWorks cabinets and shelves.

firmware executive

See *EXEC*.

Firmware Licensing System

See *FLS*.

FLS

Firmware Licensing System. The FLS enables a user to enter license keys and activate licensed features (such as RAIDsets) on the SWXRC-04 controller.

flush

To write cache data to storage media.

FMU

Fault Management Utility. The FMU controls certain spontaneous error displays to a maintenance terminal. FMU also provides for interactive review of last failure and memory system failure information.

FRU

Field replaceable unit. An FRU is the lowest-level hardware component that can be replaced by field personnel.

FWD SCSI

Fast, wide, differential SCSI. This differential SCSI bus has a 16-bit data transfer rate of up to 20 MB/s. See also FD SCSI and SCSI.

SWXSS-01 controller shelf

The StorageWorks controller shelf used for SWXSC-series controller modules, cache modules, and shelf power units.

SWXSC-AA storage enclosure

A StorageWorks shelf that can contain an entire subsystem, including storage devices, power supplies, and controllers. It is normally housed in an SWXSC-AA cabinet.

SWXSS-02 SBB shelf

A StorageWorks shelf used for only power units and SBBs.

host

The primary or controlling computer to which a storage subsystem is attached.

host logical unit

See *logical unit*.

host terminal

See *virtual termina*l.

hot swap

A method of device replacement whereby the complete system remains on line and active during device removal and reinstallation. The device being removed or reinstalled is the only device that cannot perform operations during this process.

IBR

Initial boot record. The IBR is a table of information placed in memory in accordance with i960 processor specifications. The i960 reads the IBR during initialization in order to configure itself, as well as position the program counter at the start of core MIST.

initial boot record

See *IBR*.

initiator

The SCSI bus member that requests an operation to be performed by another member (the target). When an SWXRC-04 controller interacts with its physical storage devices, it is the initiator. Another example is when the host CPU interacts with the SWXRC-04 controller, the host is the initiator.

instance code

The four-byte value transmitted in the error log packet that is key to interpreting the error.

KILL line

The controller-to-controller disable signal used in a dual- redundant configuration.

last failure

Last failure refers to event/error information specifically generated upon the sudden termination of executing firmware. For example, last failure occurs when there is an abrupt controller failure (such as when power is removed).

last failure logging

See *LFL*.

least recently used

See *LRU*.

LFL

Last failure logging. LFL is the spontaneous posting of last failure information to a connected terminal. The LFL display enables the user to read last failure information without invoking a separate utility for translation.

logical unit

Also called a host logical unit or simply a unit. A logical unit is a device or group of devices addressable as one unit by the host. (A logical unit is not necessarily a *LUN*.)

logical unit number

See *LUN*.

LRU

Least recently used. LRU is the block replacement algorithm for the read cache.

LUN

Logical unit number. A LUN is a value from 0-7 that identifies a subset of a SCSI target to a SCSI initiator. LUNs use their target's bus connection to communicate on the SCSI bus.

maintenance terminal

The operator terminal used to identify a controller, to enable its host paths, to define its subsystem configuration, and to check its status. The maintenance terminal interface is designed to accept any terminal conforming to EIA-423. A maintenance terminal is only required to configure a storage subsystem and is not required for normal operations.

metadata

Special data written to a disk device and inaccessible to the host CPU. Metadata improve error detection and media defect management for the disk device.

MIST

Module integrity self-test. MIST tests controller functions upon initialization. See also *DAEMON*.

module integrity self-test

See *MIST*.

node

An intelligent entity in a distributed computing configuration. Nodes are independent but linked, as in a network or a cluster, becoming parts of a whole.

nonredundant

A configuration in which there is no backup hardware in place for the hardware that is present.

nontransportable

An operator device assignment that indicates the device contains metadata. Devices assigned as nontransportable can be moved among SWXRC-04 controller subsystems, but cannot be moved directly to non-SWXRC-04 controller systems. See also transportable.

nonvolatile

See *NV*.

nonvolatile memory

See *NVMEM*.

nonvolatile parameter memory

See *NVPM*.

NV

Nonvolatile. NV describes memory, the contents of which survive loss of power.

NVMEM

Nonvolatile memory. NVMEM is the battery backed-up SRAM on the controller module.

NVPM

Nonvolatile parameter memory. NVPM is a portion of NVMEM used to store controller configuration data.

OCP

Operator control panel. The OCP is the control/indicator panel associated with a device. The OCP is usually mounted on the device and is accessible to the operator.

offline

One of the possible status conditions of a mass storage device or server. When a device is offline, it is not capable of communicating with the controller. When the controller is offline, it is inaccessible to any node in the configuration.

operator control panel

See *OCP*.

PCMCIA

Personal Computer Memory Card Industry Association. PCMCIA is an organization that develops standards for ROM memory cards.

Personal Computer Memory Card Industry Association

See *PCMCIA*.

port

The hardware and software used to connect a controller to a communication bus, such as a SCSI bus.

port/target/LUN

See *PTL*.

program card

The PCMCIA card containing the SWXRC-04 controller operating firmware.

PTL

Port/target/LUN. PTL is a three-number hierarchical value representing a device location to a SCSI initiator. For example, PTL 143 is a device on port 1 of the initiator, target 4 on port 1, and LUN 3 under target 4.

qualified device

A device that has been fully tested in all appropriate StorageWorks hardware and software configurations, and is in complete compliance with DIGITAL and country-specific standards (for example, FCC and TÜV).

quiesce

To make a bus inactive or dormant. The operator must quiesce SCSI bus operations, for example, during a device warm swap.

radio frequency interference

See *RFI*.

RAID

Redundant array of independent disks. RAID is a set of storage techniques devised to increase the performance and availability of a storage subsystem.

RAIDset

Three to fourteen physical disks configured as a container. RAIDsets enable one physical disk to fail without loss of data.

read cache

A block of high-speed memory used by a controller to buffer data being read from storage devices by a 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. The controller maintains in the cache copies of data recently requested by the host, and may fetch blocks of data ahead in anticipation that the controller accesses the next sequential blocks. In a normal read cache, host write requests are handled as usual, without involving the caching mechanism. See also write through cache.

reconstruction

The process of regenerating the data from one failed RAIDset member. Reconstruction is possible through XORing the surviving members' data with the parity data (or recalculating parity data, if a parity member failed). When the missing data is reconstructed it is placed on a disk from the spareset, if one is available. If there is no spare disk, the RAIDset goes reduced and does not reconstruct.

reduced

A RAIDset that has one failed member is said to be running reduced. RAID functions enable the missing data from the failed member to be reconstructed and accessed on an as-needed basis, using the surviving members' data and parity data. A RAIDset does not run reduced (or at all) if more than one member has failed.

Redundant Array of Independent Disks

See *RAID*.

replacement

The process of changing a spareset device into an active member of a RAIDset (replacing a failed RAIDset device). Spareset members are selected as replacements based on either "best fit" or "best performance." Data previously backed up on tape is retrieved for disk storage using the normal priority. Backup is used to preserve information in the event of a disk failure. Restore is used to recover the information.)

replacement policy

The firmware-controlled method by which a spare disk (from the spareset) is selected to replace a disk that has failed in a RAIDset.

restore

Data previously backed up on tape is retrieved for disk storage using the normal priority. Backup is used to preserve information in the event of a disk failure. Restore is used to recover the information.

RFI

Radio frequency interference. RFI is the impairment of a signal by an unwanted radio signal or radio disturbance.

SBB

StorageWorks building block. An SBB is a device housed in a standard StorageWorks SBB carrier. An SBB has a standard physical and electrical interface that is compatible with those of StorageWorks shelves and enclosures.

SBB shelf

StorageWorks building block shelf. This is a StorageWorks shelf, such as the BA350-Sx, designed to house plug-in SBB modules.

SCSI

Small Computer System Interface. SCSI is an ANSI interface defining the physical and electrical parameters of a parallel I/O bus used to connect hosts to a maximum of seven devices. The StorageWorks device interface is implemented according to the SCSI-2 standard, enabling the synchronous transfer of 8-bit data at rates of up to 10 MB/s.

SCSI device

A host computer adapter, a peripheral controller, or a peripheral device that can be attached to the SCSI bus.

SCSI device ID

Also referred to as a *target* ID. This ID is the physical address an initiator uses to connect with a target. Each target is assigned a unique target address.

SCSI-A cable

A 50-conductor, 25 twisted-pair cable used for single- ended, SCSI bus connections.

SCSI-P cable

A 68-conductor, 34 twisted-pair cable used for differential bus connections.

shelf brackets

Sheet metal components designed to attach and position StorageWorks shelves in their associated enclosures.

signal converter

Also called an *adapter*. A signal converter converts the protocol and hardware interface of one bus type into that of another without changing the function of the bus.

single cabinet power configuration

A cabinet ac power configuration in which only one ac source and one ac power supply is used to supply dc power to the cabinet's SBB shelves.

skirt

A trim panel designed to mount around the base of a cabinet.

Small Computer System Interface

See *SCSI*.

software product description

See *SPD*.

spareset

A pool of disk drives from which a controller can draw to replace failed members of a RAIDset.

SPD

Software product description. The SPD describes the function of a program.

storageset

A grouping of disk drives that make up a new distinct container. Stripesets and RAIDsets are examples of storagesets.

StorageWorks

DIGITAL's family of modular data storage products that enables customers to design and configure their own storage subsystems. Components include power, packaging, cabling, devices, controllers, and software. Customers can integrate devices and array controllers in StorageWorks enclosures to form storage subsystems.

StorageWorks building block

See *SBB*.

stripeset

A virtual disk drive with its physical data spread across from 2-14 physical disks. Stripeset configurations do not include a data recovery mechanism.

supported device

A device tested as functionally compatible with an approved StorageWorks hardware and software configuration.

surviving controller

The controller in a dual-redundant pair that assumes service to its companion's devices when the companion fails. See also failover.

tagged command queuing

A technique that enables a device to have multiple I/O requests outstanding to it at one time.

target

A member of a SCSI bus that carries out operations requested by an initiator. An example is the SWXRC-04 controller, which is a target of its host CPU.

target ID

Also referred to as *SCSI device ID.* This ID is the physical address an initiator uses to connect with a target. Each target is assigned a unique target address.

template

The group or type to which an error event log to the host belongs. Template type determines how the EIP is formatted.

transportable

An operator device assignment that indicates the device does not contain metadata. Transportable devices can be moved between SWXRC-04 controller subsystems and non-SWXRC-04 controller systems. However, such devices do not support forced error, and should not be set to transportable after correct installation in an SWXRC-04controller subsystem. See also *nontransportable*.

unit

See *logical unit*.

unwritten cache data

Data suspended in write-back cache memory which has not been written to storage media yet, even though the host operation using the data has completed.

value-added firmware

The firmware that provides optional, licensed features such as logical block mapping, cache, RAID, and so on.

virtual terminal

A software path from an operator terminal on the host to the controller's CLI interface. The path can be established via the maintenance port on the controller. A virtual terminal is also sometimes called a host terminal.

warm swap

A controller function that enables devices to be added, removed, or replaced while the subsystem remains operational. All activity on the device's SCSI bus must normally be halted for the duration of the warm swap operation.

write hole

Undetectable RAID level 5 data corruption. A write hole is caused by the successful writing of cached data but not the data parity, or vice versa. Write holes occur under conditions such as power outages, where the writing of these two elements can be abruptly interrupted. A battery backed-up cache design eliminates the write hole, because data is preserved and writes can be retried.

write-back cache

A cache configuration that increases the 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.

write-through cache

A technique for handling host write requests in read caches. When the host requests a write operation, the cache writes data directly to the external storage device and updates the cache memory to ensure that the memory does not contain obsolete data. This technique increases the chances that future host read requests can be filled from the cache. The host sees the write operation as complete only after the external storage device has been updated. See also read *cache*.

Index

This chapter provides an index of topics covered in this manual.