COMPAQ
STORAGEWORKS

HSZ70 Array Controller
HSOF Version 7.3

EK–HSZ70–CG. B01

Compaq Computer Corporation
Houston, Texas

Configuration Manual

January 1999

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the manuals, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense. Restrictions apply to the use of the local-connection port on this series of controllers; failure to observe these restrictions may result in harmful interference. Always disconnect this port as soon as possible after completing the setup operation. Any changes or modifications made to this equipment may void the user's authority to operate the equipment.

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

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

Avertissement!
Cet appareil est un appareil de Classe A. Dans un environnement résidentiel cet appareil peut provoquer des brouillages radioélectriques. Dans ce cas, il peut être demandé à l' utilisateur de prendre les mesures appropriées.

Contents

Contents

Chapter 2 Planning a Subsystem

Chapter 3 Planning Storage

Chapter 4 Subsystem Configuration Procedures

Chapter 5 Periodic Procedures

Appendix A   Controller Specifications

Appendix B   System Profiles

Glossary

Index

Figures

Tables

Preface

This book describes the features of the HSZ70 array controller and configuration procedures for the controller and storagesets running HSOF Version 7.3.

This book does not contain information about the operating environments to which the controller may be connected, nor does it contain detailed information about subsystem enclosures or their components. See the documentation that accompanied these peripherals for information about them.

Precautions

Use the precautions described in the following paragraphs when you are carrying out any servicing procedures:

■ Electrostatic Discharge Precautions, page xiii

■ VHDCI Cable Precautions, page xiv

■ Local-Connection Maintenance Port Precautions, page xiv

Electrostatic Discharge Precautions

Static electricity collects on all nonconducting material, such as paper, cloth, and plastic. An electrostatic discharge (ESD) can easily damage a controller or other subsystem component even though you may not see or feel the discharge. Follow these precautions whenever you're servicing a subsystem or one of its components:

■ Always use an ESD wrist strap when servicing the controller or other components in the subsystem. Make sure that the strap contacts bare skin and fits snugly, and that its grounding lead is attached to a bus that is a verified earth ground.

■ Before touching any circuit board or component, always touch a verifiable earth ground to discharge any static electricity that may be present in your clothing.

■ Always keep circuit boards and components away from
  nonconducting material.

■ Always keep clothing away from circuit boards and components.

■ Always use antistatic bags and grounding mats for storing circuit
  boards or components during replacement procedures.

■ Always keep the ESD cover over the program card when the card is
  in the controller. If you remove the card, put it in its original
  carrying case. Never touch the contacts or twist or bend the card
  while handling it.

■ Do not touch the connector pins of a cable when it is attached to a
  component or host.

VHDCI Cable Precautions

All of the cables to the controller, cache module, and external cache
battery use very-high-density cable interconnect connectors (VHDCI).
These connectors have extraordinarily small mating surfaces that can
be adversely affected by dust and movement.

Use the following precautions when connecting cables that use VHDCI
connectors:

■ Clean the mating surfaces with a blast of clean air.

■ Mate the connectors by hand, then tighten the retaining screws to
  1.5 inch-pounds—approximately 1/4 additional turn after the
  connectors have fully mated.

■ Test the assembly by gently pulling on the cable, which should not
  produce visible separation.

Local-Connection Maintenance Port Precautions

The local-connection port generates, uses, and radiates radio-frequency
energy through cables that are connected to it. This energy may
interfere with radio and television reception. Do not leave a cable
connected to this port when you're not communicating with the
controller.

Conventions

This book adheres to the typographical conventions and special notices found in the table and paragraphs that follow.

Typographical Conventions

Convention	Meaning
ALLCAPS BOLD	Command syntax that must be entered exactly as shown, for example: **SET FAILOVER COPY=OTHER_CONTROLLER**
ALLCAPS	Command discussed within text, for example: "Use the SHOW SPARESET command to show the contents of the spareset."
Monospaced	Screen display.
Sans serif italic	Command variable or numeric value that you supply, for example: SHOW *RAIDset-name* or **SET THIS_CONTROLLER ID=***(n,n,n,n,)*
italic	Reference to other books, for example: "See *HSZ70 Array Controller HSOF Version 7.3 Configuration Manual* for details."
⋮	Indicates that a portion of an example or figure has been omitted. **ADD RAIDSET RAID1 DISK10000 DISK20000 INITIALIZE RAID1** . . . **SHOW RAID1**
"this controller"	The controller serving your current CLI session through a local or remote terminal.
"other controller"	The controller in a dual-redundant pair that's connected to the controller serving your current CLI session.

Special Notices

This book does not contain detailed descriptions of standard safety procedures. However, it does contain warnings for procedures that could cause personal injury and cautions for procedures that could damage the controller or its related components. Look for these symbols when you're carrying out the procedures in this book:

Warning A warning indicates the presence of a hazard that can cause personal injury if you do not avoid the hazard.

Caution A caution indicates the presence of a hazard that might damage hardware, corrupt software, or cause a loss of data.

Tip A tip provides alternative methods or procedures that may not be immediately obvious. A tip may also alert prior customers that the controller's behavior being discussed is different from prior software or hardware versions.

Note A note provides additional information that's important to the completion of an instruction or procedure.

Required Tools

The following tools are needed for servicing the controller, cache module, and external cache battery:

■ A small screwdriver for loosening and tightening the cable-retaining screws.

■ An antistatic wrist strap.

■ An antistatic mat on which to place modules during servicing.

■ An SBB Extractor for removing StorageWorks building blocks. This tool is not required, but it will enable you to provide more efficient service.

Related Publications

The following table lists some of the documents related to the use of the controller, cache module, and external cache battery.

Document Title	Part Number
HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual	EK–CLI70–RM. B01
HSZ70 Array Controller HSOF Version 7.3 Configuration Manual	EK–HSZ70–CG. B01
HSZ70 Array Controller HSOF Version 7.3 Service Manual	EK–HSZ70–SV. B01
HSZ70 Family Array Controller Operating Software (HSOF) Version 7.3 Software Product Description	SPD *xx.xx*. 00
Getting Started–HSZ70 Solutions Software Version 7.3 for DIGITAL UNIX	AA–R60KD–TE
Getting Started–HSZ70 Solutions Software Version 7.3 for OpenVMS	AA–R8A7D–TE
Polycenter Console Manager	See the *Getting Started* guide for the platform-specific order number
StorageWorks Array Controller HSZ70 Array Controller Operating Software HSOF Version 7.3 Release Notes	EK–HSZ70–RN. B01
StorageWorks Getting Started with Command Console, Version 2.1	AA–R0HJC–TE
StorageWorks Ultra SCSI RAID Cabinet Subsystem (SW600) Installation and User's Guide	EK–SW600–UG
StorageWorks Ultra SCSI RAID Enclosure (BA370-Series) User's Guide	EK–BA370–UG
The RAIDBOOK—A Source for RAID Technology	RAID Advisory Board

Revision History

The current revisions of this document include:

EK-HSC70-CG.B01   HSOF V7.3   December 1998
EK-HSZ70-CG.A01   HSOF V7.0   July 1997

C H A P T E R 1

Subsystem Introduction

This chapter of the HSZ70 Array Controller Configuration Manual introduces the features and components of the HSZ70 controller:

Typical Installation

Figure 1–1 shows the major components, or basic building blocks, of a typical storage subsystem. Table 1–1 lists and describes these building blocks.

Figure 1–1   Subsystem Building Blocks
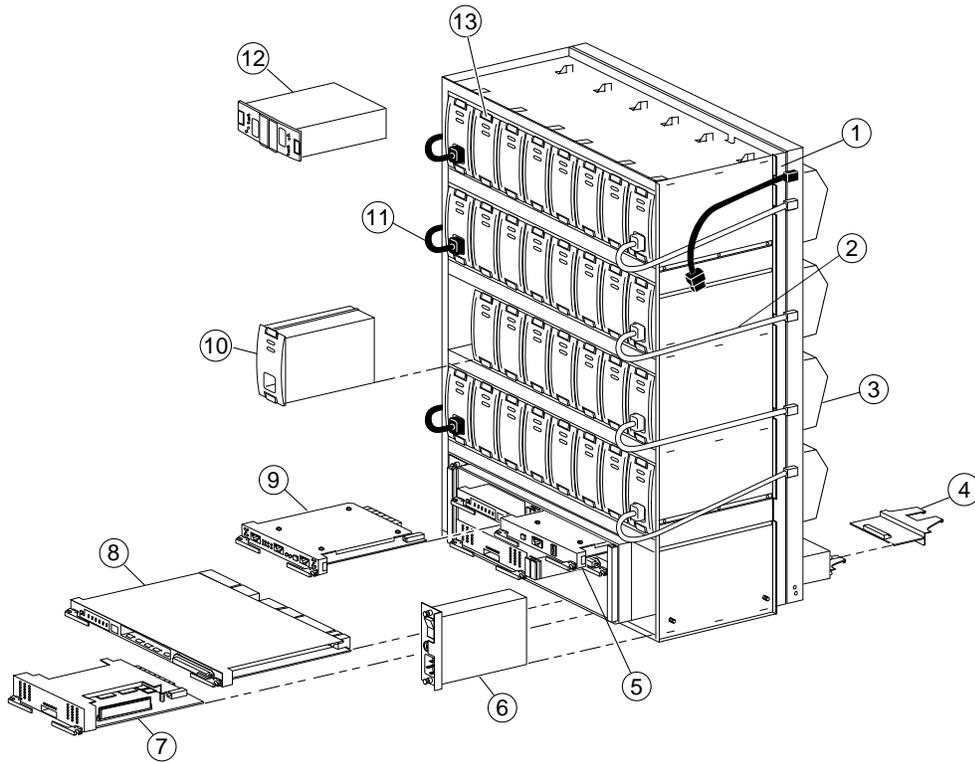
CXO6702A

Table 1–1   Key to Figure 1–1

Item	Description	Part No.
1	BA370 rack-mountable enclosure	DS–BA370–AA
2	Power cable kit (white)	17-03718-09
3	Cooling fans; 8 (2 per shelf)	DS–BA35X–MK
4	I/O module; 6	DS–BA35X–MN
5	PVA module (provides a unique address to each enclosure in an extended subsystem).	DS–BA35X–EC

Table 1–1   Key to Figure 1–1 (Continued)

Item	Description	Part No.
6	AC input module	DS–BA35X–HE
7	Cache module; 1 or 2	70-33256-01
8	HSZ70 array controller; 1 or 2	DS-HSZ70-AH
9	Environmental monitoring unit (EMU) The EMU monitors the subsystem environment alerting the controller of equipment failures that could cause an abnormal environment.	DS–BA35X–EB
10	180-watt power supply; 8 (2 per shelf)	DS–BA35X–HH
11	Power cable kit (black)	17-03718-10
12	■ External cache battery (ECB), single	DS–HS35X–BC
	■ External cache battery (ECB), dual (two ECBs in one Storage Building Block (SBB), which provide backup power to the cache modules during a primary power failure.	DS–HS35X–BD
13	Subsystem Building Block (SBB) - a tape or disk drive unit inside a standard case.	See release notes for qualified disk/ tape drive numbers

Summary of HSZ70 Features

Table 1–2 contains a summary of the HSZ70 Array Controller features.

Table 1–2   Summary of Controller Features

Feature	Supported
Host protocol	SCSI–2
Host bus interconnect	Wide Ultra Differential SCSI–2
Device protocol	SCSI–2
Device bus interconnect	Fast Wide Ultra Single-ended SCSI-2

Table 1–2   Summary of Controller Features (Continued)

Feature	Supported
Number of SCSI device ports	6
Number of SCSI device targets per port	12
Maximum number of SCSI devices (with two additional BA370 shelves)	72
RAID levels	0, 1, 0+1, 3/5
Cache size	64 or 128 MB
Mirrored write-back cache sizes	32 or 64 MB
Max number of host target IDs per controller	8
Program card updates	Yes
Device warm swap	Yes
Exercisers for testing disks	Yes
Tape drives, loaders, and libraries	Yes
Number of configuration entities (devices + storagesets + partitions + units)	191
Maximum number of RAID 5 storagesets	20
Maximum number of RAID 5 and RAID 1 storagesets: ■ Dual-controller configurations ■ Single-controller configurations	30 20
Maximum number of RAID 5, RAID 1, and RAID 0 storagesets	45
Maximum number of partitions per storageset or disk drive	8
Maximum number of units presented to host	64
Maximum number of units presented to host with StorageWorks Command Console	63
Maximum number of devices per unit	32
Maximum host port transfer speed	20 MHz
Largest device, storageset, or unit	512 GB

HSZ70 Basic Description

Your controller is the intelligent bridge between your host and the devices in your storage subsystem.

The controller may be thought of as containing three main (but distinct) parts. The array controller is best understood if these three functional divisions are conceived as separate entities. The three entities are:

■   Device-side logic

■   Intelligent bridge

■   Host-side logic

The host-side logic and device-side logic sections both use SCSI-2 protocol. This means that the terms used to describe both interfaces are the same: both have ports, targets, and LUNs. This terminology can cause confusion between device-side and host-side functions. Keeping this simple model in mind can help eliminate confusion.

The model is shown in Figure 1–2, and is explained in the following sections.

Figure 1–2   Host and Storage Subsystem Bridge
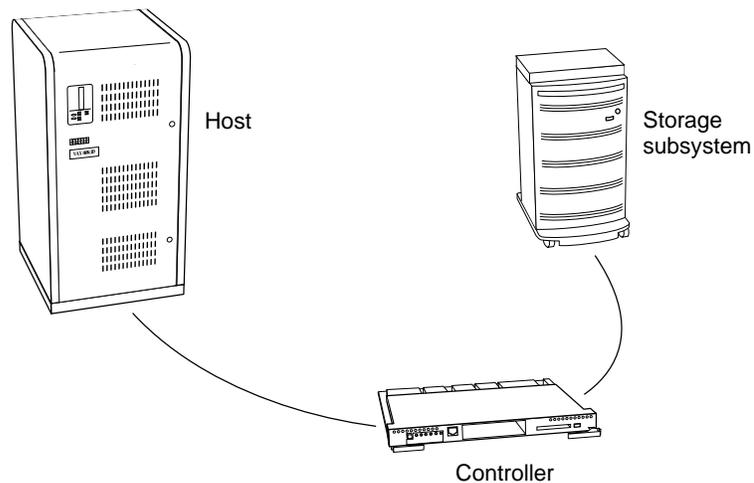
CXO5505A

Device-Side Logic

The device side is the controller's interface to a collection of physical storage devices (the device array). Each of these storage devices is capable of SCSI-2 level communication and each functions independently of the others. The controller connects to these devices through six device-side SCSI buses.

The intelligent bridge performs the internal functions of the controller. The intelligent bridge both *separates* and *links* the device-side and host-side logic functions. These functions are defined as follows:

■   Separate—The intelligent bridge is a barrier between the device-side and host-side logic, making the device-side logic (and hence the storage devices) completely invisible and inaccessible to the host.

■   Link—The intelligent bridge creates virtual disks that reference (maps), the contents of the real, device-side disks in defined ways. The bridge creates an illusion for the host: the host sees only the virtual storage devices created by the intelligent bridge and knows nothing at all about the physical storage devices that reside behind the controller (device-side). These virtual storage devices look and operate differently than the physical, device-side storage devices.

The virtual storage devices appear and act to the host as real storage devices with normal operating parameters. The host performs SCSI commands to these virtual storage devices set up by the controller. When the host sends a command to one of the virtual storage devices, the controller translates and executes the command on the physical storage device that is mapped to the virtual.

Host-Side Logic

The host-side logic is the HSZ70 Array Controller interface to one or more hosts. A host is a computer attached to one of the host-side SCSI buses. The host can access the virtual disks created by the intelligent bridge when attached to one of the host-side SCSI buses. The controller is said to "present" these virtual storage devices to the host.

The host sees only what the intelligent bridge lets it see: a number of high-capacity, high-availability, and/or high-reliability virtual storage devices. The physical reality of what is there (an array controller and an array of physical disk and/or tape drives), is completely masked.

See the product-specific release notes that accompanied the software release for the most recent list of supported devices and operating systems.

In addition to managing SCSI I/O requests, the controller provides the ability to:

■   Combine several disk drives into a single, high-performance storage unit called a storageset.

■   Partition the single disk drive or storageset.

Storagesets and Containers

Storagesets are implementations of RAID technology (Redundant Array of Independent Disks). RAID technology ensures that every unpartitioned storageset (regardless of the number of disk drives), appears to the host as a single storage unit.

See Chapter 3, "Planning Storage," for more information about storagesets and how to configure them.

The Storageset types used by the HSZ70 are shown in Figure 1–3 "Example Containers" and described in the following list:

■   RAID 0 (Stripesets)—disk drives combined in serial to increase transfer or request rates.

■   RAID 1 (Mirrorsets)—disk drives combined in parallel to provide a highly-reliable storage unit.

■   RAID 0+1 (Striped Mirrorsets)—combined mirrorsets in serial and parallel providing the highest throughput and availability.

■   RAID 3/5 (RAIDsets)—disk drives combined in serial (as RAID 0) but also store parity data ensuring high reliability.

Storagesets are augmented by two other disk storage configurations:

■   Single disk devices (also known as JBOD—just a bunch of disks). This configuration is a single disk device not a part of any of the RAID technology (storagesets). In this configuration, a single disk may be a distinct unit number.

■   Partitions—A configuration of single disk drives or a storageset whose data storage area is partitioned.

Note   Any of the storagesets shown may also be partitioned unless you are operating with a dual-redundant controller configuration.

For a complete discussion of RAID, refer to *The RAIDBOOK—A Source Book for Disk Array Technology.*

Figure 1–3 illustrates the concept of the installed disk drives being configured as storagesets, individual storage devices, or partitions. The collective term for these is "containers".

Figure 1–3   Example Containers
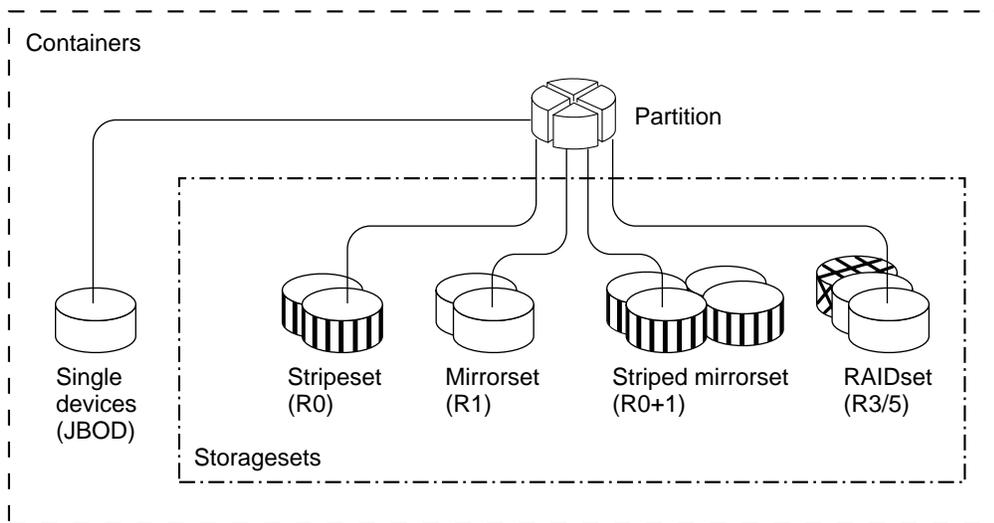
CXO6677A

The controller also allows tape drives, loaders, and libraries to be added your subsystem to meet your storage requirements.

Physically Connecting the Host to the Storage Array

The controller provides a host system physical access to a peripheral device storage array by using host and device buses. Figure 1–4 shows that the host and device bus interfaces are both Small Computer System Interface (SCSI) buses.

Figure 1–4   Subsystem Bus Block Diagram
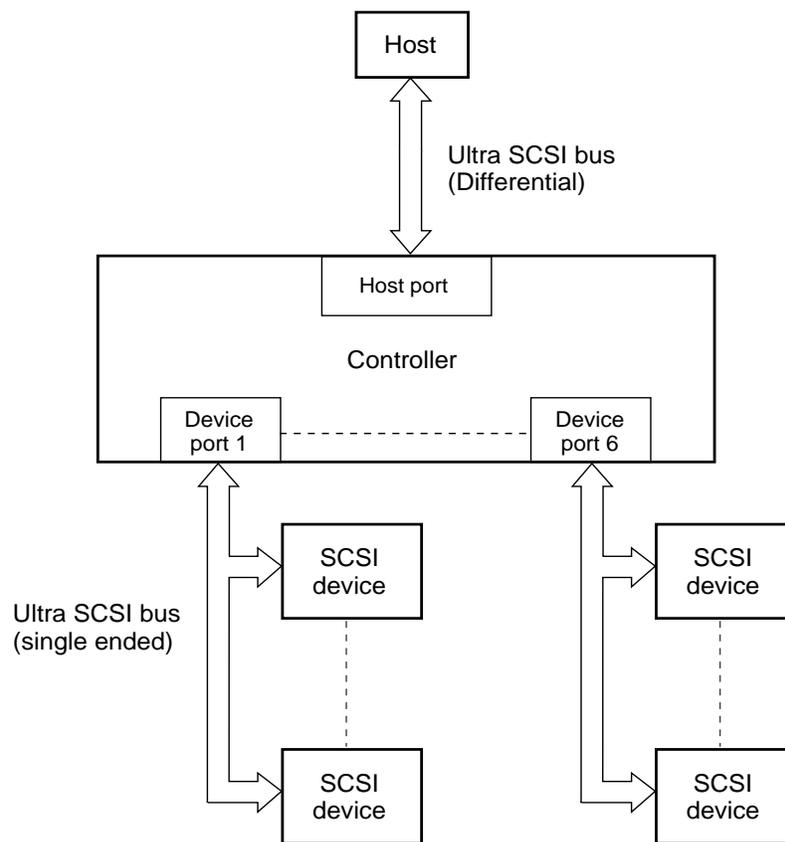
CXO4602A

Logically Connecting the Storage Array to the Host

The controller uses a two-level mapping process to logically connect the host to the storage array:

■   The controller maps the physical devices on its six device buses to storage containers created by the user.

■   The controller maps its internal containers to user-created logical units that are directly accessible by the host.

Mapping the Physical Devices

Figure 1–5 shows a typical physical storage device interface for a one-cabinet subsystem (or the first cabinet of an extended subsystem). Each of the controller's six input/output device ports supports a SCSI-2 bus connected to storage devices (the maximum number dependent upon the cabinet configuration and the SCSI specifications).

This connection is hardwired through the port I/O modules and the enclosure backplane wiring.

Controller port-target-LUN (PTL) addressing is the process by which the controller selects storage space within the specific, physical, storage device. This process consists of three steps:

■ The controller I/O port selection. The controller selects one of the six I/O ports of the controller.

■ The controller SCSI-2 bus target selection. The controller selects the device's target ID.

■ The controller-device logical unit (LUN) selection. The controller selects the desired LUN within the specific storage device. In the current implementation, there is only one LUN on each storage device, therefore that LUN address is always 0.

Note   The exception to this rule occurs when certain multi-LUN tape loaders are used (therefore they would have a LUN of 0 or 1).

Refer to Figure 1–5; the illustration depicts the PTL addressing for a single cabinet full of disk drives. The numbers in each "cell" of the storage cabinet reflect the PTL of that particular device. In a three-cabinet configuration the controller SCSI-2 target IDs would be as follows:

■ 1st cabinet: 0, 1, 2, 3

■ 2nd cabinet: 8, 9, 10, 11

■ 3rd cabinet: 12, 13, 14, 15

Note   SCSI Target IDs 4, 5, 6, and 7 are not used by the storage devices. They are reserved for the two possible controllers (ID 7, ID 6) while the other two are not used.

Figure 1–5   Physical Storage Device Interface
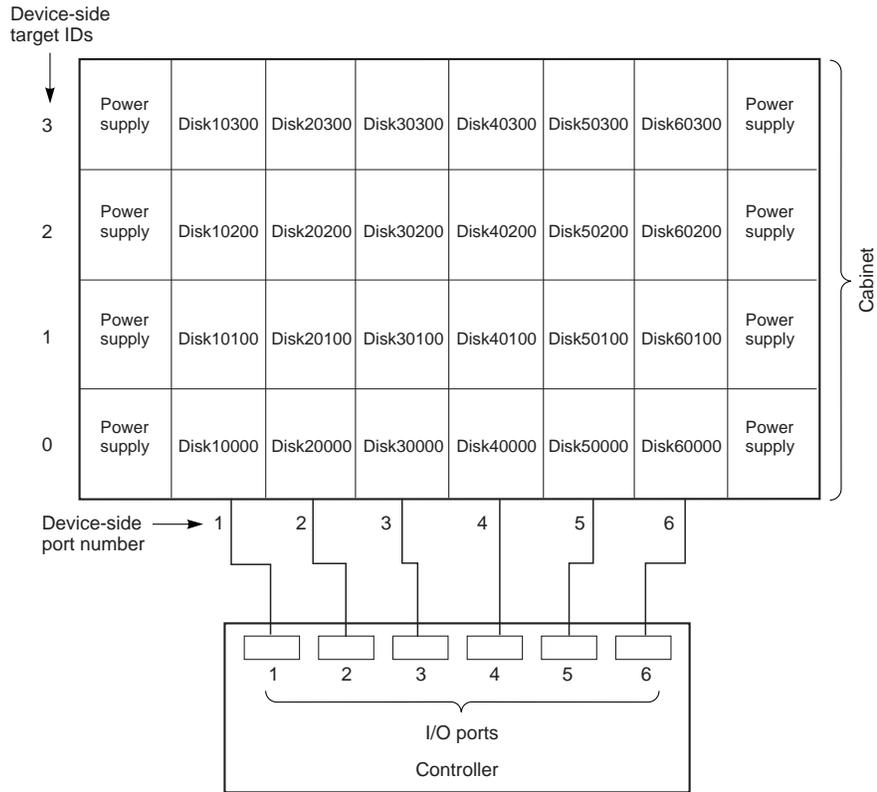
The PTL addressing of the storage devices is handled internally by the controller. The controller receives a Unit Number from the host when a read or write operation is requested. The controller translates the Unit Number into the physical PTL address to select the drive. The unit number-to-PTL correlation comes from the user-entered CLI commands:

■   ADD DISK *container-name SCSI PTL* (assigns a container name to the PTL address).

■   ADD UNIT *unit-number container-name* (assigns a Unit Number (LUN) to the container name).

For both of these commands see the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details.

Mapping the Containers

Each logical unit the controller makes available to the host is associated with a Target ID on the host SCSI bus. Figure 1–6 shows the host view of the logical storage units.

The controller offers up to eight SCSI Target IDs to the host bus. Each SCSI Target ID can offer up to eight logical units (LUNs) to the host. Each logical unit is separately addressable and has configurable operating attributes based on the container type that it is a part of.

The controller software enables the user to configure the LUNs in a variety of ways to optimize for specific performance and availability needs. The details of the configuration of each logical storage unit are transparent to the host. The host uses the performance and availability benefits of each unit without any of the overhead associated with the optimization process.

Note   Host-side SCSI LUNs and device-side SCSI LUNS share the same name but have no relationship to one another.

Each host on the storage subsystem, is connected to the SCSI-2 bus by way of a host bus adapter. The HSZ70 controller (or dual-redundant controller pair) also connects to this SCSI-2 bus through the controller host ports.

The SET *this_controller ID=* command selects up to eight host target IDs to one controller (or controller pair). Figure 1–6 shows an example of a subsystem using host target IDs of 0, 1, 2, 3, 4, 5, 8, and 9 (from the range of 0—15). In each of the storage disks shown in the illustration, the logical storage unit number is also noted. These logical unit numbers (LUNs) were assigned with the ADD UNIT *unit-number container-name* command.

Note   In a typical subsystem, one of the LUNs is reserved for the StorageWorks Command Console (SWCC).

Figure 1–6   Example Host View of Target Addressing

```
┌──────────┐
│   Host   │
│ adapter  │──┐
│  at ID6  │  │
└──────────┘  │
┌──────────┐  │   SCSI bus
│   Host   │  │
│ adapter  │──┘                                              ~~
│  at ID7  │ ID0          ID1          ID2          ID3          ID9
└──────────┘ │            │            │            │            │

  ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐
  │  D0   │    │ D100  │    │ D200  │    │ D300  │    │ D900  │
  └───────┘    └───────┘    └───────┘    └───────┘    └───────┘

  ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐
  │  D1   │    │ D101  │    │ D201  │    │ D301  │    │ D901  │
  └───────┘    └───────┘    └───────┘    └───────┘    └───────┘

  ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐
  │  D2   │    │ D102  │    │ D202  │    │ D302  │    │ D902  │
  └───────┘    └───────┘    └───────┘    └───────┘    └───────┘
     ·            ·            ·            ·            ·
     ·            ·            ·            ·            ·
     ·            ·            ·            ·            ·
  ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐    ┌───────┐
  │  D7   │    │ D107  │    │ D207  │    │ D307  │    │ D907  │
  └───────┘    └───────┘    └───────┘    └───────┘    └───────┘
```
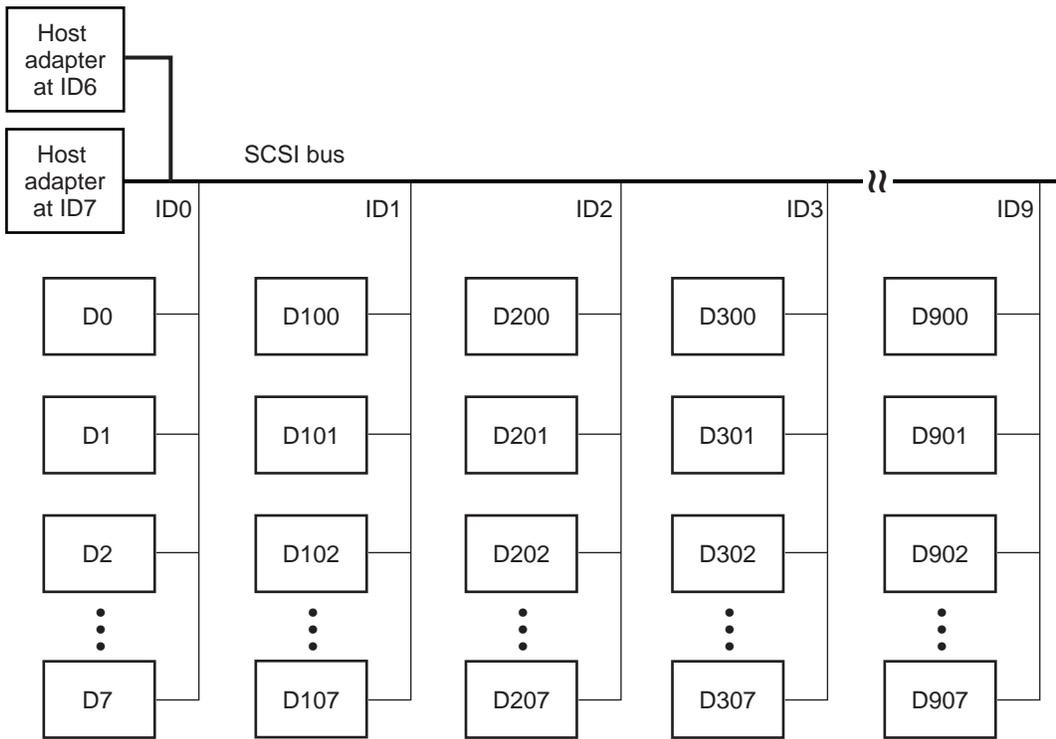
CXO6700A

Controller Key Components

The HSZ70 is made up of the key components shown in Figure 1–7.

Figure 1–7   Key Controller Components
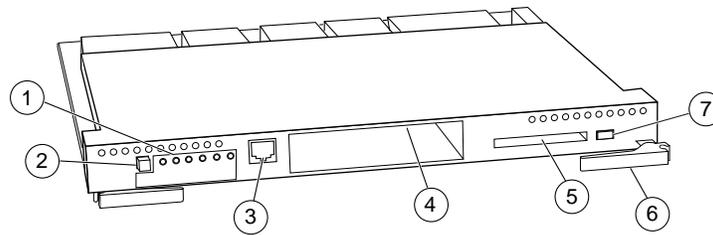
CXO5503D

Table 1–3   Key to Figure 1–7

Item	Description
①	Operator Control Panel (OCP)—A collection of amber-colored LEDs that indicate status of the controller (see "Operator Control Panel," page 1-15 and the *HSZ70 Array Controller HSOF Version 7.3 Service Manual* for details).
②	Reset Button/Indicator—Performs a hard reset of the controller when pushed. Its light indicates normal operation of the controller when flashing at a once per second rate.
③	Maintenance Connection Port—A connection port for a maintenance terminal or PC. This port allows communication with the controller for setup and configuration.
④	Host Connection Port—A port for the connection of the host CPU with the controller.
⑤	Program Card Slot—A place to insert the PCMCIA program card. The card **must** be in place for normal operation. Each time the controller reboots, the HSOF software is read from the program card.
⑥	Program Card Eject Button—Ejects the PCMCIA program card from the program card slot when pushed.
⑦	Locking Levers—Levers to lock the controller in the shelf.

Under normal circumstances, you will not need to remove the controller from its cabinet. The components that you will use most often are located on the front of the controller.

The enclosure backplane enables two controllers to communicate with each other in dual-redundant configurations. It also contains the device ports that enable the controller to communicate with the devices in the subsystem.

Each controller is supported by its own cache module. Use Figure 1–8 in conjunction with Figure 1–1 to locate the installed position of the cache module-to-controller support in a BA370 rack-mountable enclosure.

Figure 1–8   Location of Controllers and Cache Modules

EMU	PVA
Controller A	
Controller B	
Cache module A	Cache module B
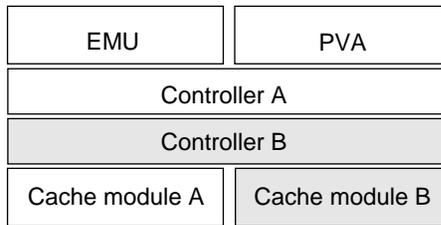
CXO6283A

Tip   For single controller installations, it is recommended that you use the slots for controller A and cache module A. Slot A responds to SCSI target ID number 7; slot B responds to target ID number 6.

Operator Control Panel

The operator control panel (OCP; see Figure 1–9) contains a ① reset button, ② six port quiesce buttons, and ③ six LEDs:

■   The Reset button normally flashes at a once per second rate indicating that the controller is functioning properly.

■   The Port quiesce buttons are used to turn off the I/O activity on the controller device ports. To quiesce a port, push its port button and hold until the corresponding port LED remains lit. Push the port button again to resume I/O activity on the port.

■   The six LEDs correspond to the six controller device ports and remain off during normal operation. If an error occurs, the reset button and LEDs illuminate in a solid or flashing pattern to help you diagnose the problem (Appendix A, "Operator Control Panel LED Description").

Figure 1–9   Operator Control Panel

CXO6547A
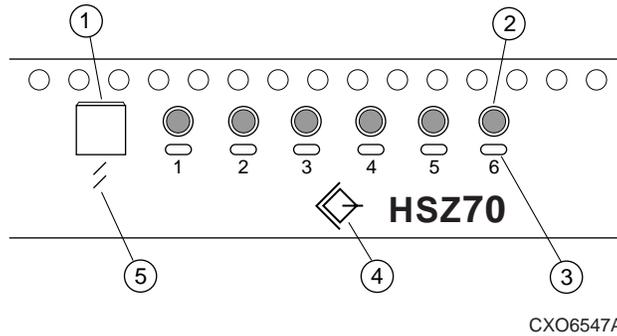
In addition, there are two international symbols placed on the front of the OCP:

■   ④ = SCSI Standard symbol for differential SCSI bus. This identifies the HSZ70 Array Controller as a SCSI-2 differential device to the host.

■   ⑤ = ISO 7000 Standard symbol for "reset" (or initialization). This symbol is placed just below the controller reset button.

See Figure 1–7 for the location of the OCP on the HSZ70 Array Controller and the *HSZ70 Array Controller HSOF Version 7.3 Service Manual* for an explanation of the LED codes that may appear on the OCP.

Once installed, configured, and running, you should periodically check the HSZ70 control panel. If an error should occur, one or more of the LED lights on the control panel flashes in a pattern that will help you diagnose the problem. Refer to the *HSZ70 Array Controller HSOF Version 7.3 Service Manual* for details about troubleshooting your controller.

Maintenance Port

You can access the controller to modify CLI commands (add a disk drive, remove a disk drive, and so on) in two ways:

■ Through a local terminal/PC via the maintenance port

■ Through a remote terminal—sometimes called a virtual terminal or host console—via the host.

It is recommended that you use a local terminal to carry out the troubleshooting and servicing procedures in this manual. See "Establishing a Local Connection to the Controller," page 4-2, for more information about connecting the array controller with a maintenance port cable.

Controller Utilities and Exercisers

The controller software includes the following utilities and exercisers to assist in troubleshooting and maintaining the controller and the other modules that support its operation:

■ Fault Management Utility (FMU)

■ Virtual Terminal Display (VTDPY)

■ Disk Inline Exerciser (DILX)

■ Configuration Utility (CONFIG)

■ HSUTIL

■ Code Load and Code Patch Utility (CLCP)

■ Clone Utility (CLONE)

■ Change Volume Serial Number Utility

■ Device Statistics Utility

■ Field Replacement Utility (FRUTIL)

Each of these is described in the paragraphs that follow.

Fault Management Utility

The Fault Management Utility (FMU) provides a limited interface to the controller fault-management system. As a troubleshooting tool, you can use the FMU to:

■ Display the last-failure and memory-system-failure entries that the fault-management software stores in controller nonvolatile memory (NVMEM).

■ Translate many of the event messages that are contained in the entries related to the significant events and failures. For example, entries may contain codes that indicate the cause of the event, the software component that reported the event, the repair action, and so on.

■ Set the display characteristics of spontaneous events and failures that the fault-management system sends to the local terminal or host.

Virtual Terminal Display

Use the virtual terminal display (VTDPY) utility to aid in troubleshooting the following issues:

■ Communication between the controller and its hosts.

■ Communication between the controller and the devices in the subsystem.

■ The state and I/O activity of the logical units, devices, and device ports in the subsystem.

Disk Inline Exerciser

Use the disk inline exerciser (DILX) to investigate the data-transfer capabilities of disk drives. DILX tests and verifies operation of the controller and the SCSI–2 disk drives attached to it. DILX generates intense read and write loads to the disk drive while monitoring the drive performance and status.

Configuration Utility

Use the configuration (CONFIG) utility to add one or more storage devices to the subsystem. This utility checks the device ports for new disk drives, then adds them to the controller configuration and automatically names them.

HSUTIL

Use HSUTIL to upgrade the firmware on disk drives in the subsystem and to format disk drives.

Code Load and Code Patch Utility

Use Code Load and Code Patch (CLCP) utility to upgrade or patch the controller software and the EMU software. Whenever you install a new controller, you must have the correct software version and patch number.

Note   Only field service personnel are authorized to upload EMU microcode updates. Contact the Customer Service Center (CSC) for directions in obtaining the appropriate EMU microcode and installation guide.

Clone Utility

Use the Clone utility to duplicate the data on any unpartitioned storageset or individual drive. Back up the cloned data while the actual storageset remains online.

Note   The clone utility may not be used with partitioned mirrorsets or partitioned Stripesets.

Change Volume Serial Number Utility

The Change Volume Serial Number (CHVSN) utility generates a new volume serial number (called *VSN*) for the specified device and writes it on the media. It is a way to eliminate duplicate volume serial numbers and to rename duplicates with different volume serial numbers.

Note   Only authorized service personnel may use this utility.

Device Statistics Utility

The Device Statistics (DSTAT) utility allows you to log I/O activity on a controller over an extended period of time. Later, you can analyze that log to determine where the bottlenecks are and how to tune the controller for optimum performance.

Note   Only authorized service personnel may use this utility.

Field Replacement Utility

Use the Field replacement Utility (FRUTIL) to replace a controller, cache module, or cache battery when operating in a dual-redundant controller configuration. FRUTIL guides you through the process of replacing these modules without interrupting the system. Enter RUN *frutil* from the maintenance terminal prompt to begin the utility. See the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for more information on this command. See the *HSZ70 Array Controller HSOF Version 7.3 Service Manual* regarding the use of the utility.

Cache Module

Each controller requires a companion cache module (Figure 1–10). Table 1–4 lists the descriptions and part numbers of the cache module. Figure 1–8 shows the location of a controller's companion cache module.

The cache module, which contains up to 128 MB of memory, increases the subsystem I/O performance by providing read, read-ahead, write-through, and write-back caching. The size of the memory contained in the cache module depends on the configuration of the SIMMs, with the supported combinations shown in Table 1–4.

Table 1–4   Cache Module Memory Configurations

SIMMs	Quantity	Total Cache Memory
32 MB	2	64 MB
32 MB	4	128 MB

Note   See "Replacing SIMMS" in the *HSZ70 Array Controller HSOF Version 7.3 Service Manual* for SIMM configuration information.

Figure 1–10 Cache Module

CXO5714A
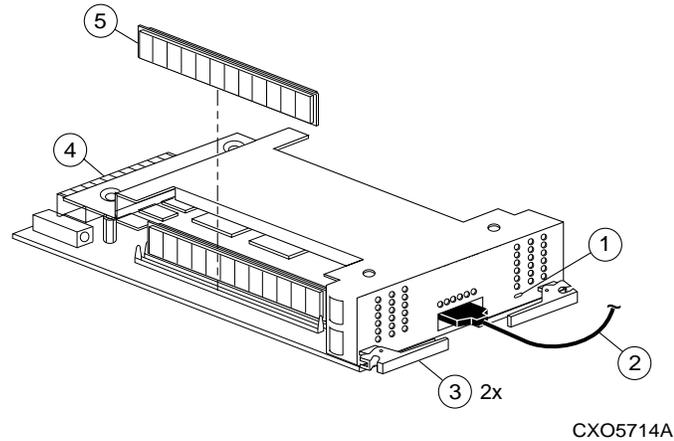
Table 1–5   Key to Figure 1–10

Item	Description
①	Cache memory power-on LED
②	ECB "Y" cable (to External Cache Battery)
③	Locking release levers (2)
④	Backplane connector
⑤	SIMM—Single Inline Memory Module (32 MB memory)

External Cache Battery

To preserve the write-back cache data in the event of a primary power failure, a cache module must be connected to an external cache battery (ECB) or an uninterruptible power supply (UPS).

There are two versions of ECBs supplied:

■ Single-battery ECB for single controller configurations.

■ Dual-battery ECB for dual-redundant controller configurations.

The dual-redundant ECB is shown in Figure 1–11. When the batteries are fully charged, an ECB can preserve 128 MB of cache memory for

24 hours. Battery capacity, however, depends upon the size of memory contained in the cache module. The ECB capacity versus battery size is listed in Table 1–6.

Table 1–6   ECB Capacity versus Battery Size

Size	SIMM Combinations	Capacity
64 MB	Two, 32 MB	96 hours
128 MB	Four, 32 MB	48 hours

Figure 1–11 ECB for Dual-Redundant Controller Configurations

CXO5713A
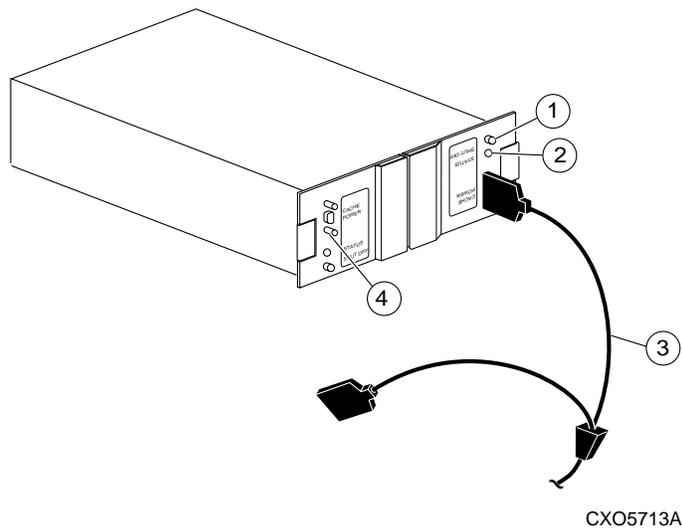
Table 1–7   Key to Figure 1–11

Item	Description
①	Battery disable switch
②	Status LED
③	ECB Y cable (to Cache Module)
④	Faceplate and controls for second battery

The HSZ70 Array Controller can operate with only 32 MB of memory on the cache when operating in non-mirrored mode. A minimum of 64 MB of memory is required on each cache module if the mirrored cache feature is to be used. The controller will not initialize when operating in mirrored cache mode and either cache module contains less than 64 MB of memory.

Charging Diagnostics

Whenever you restart the controller, its diagnostic routines automatically check the charge in the ECB batteries. If the batteries are fully charged, the controller reports them as fully charged and rechecks them every 24 hours. If the batteries are charging, the controller rechecks them every 4 minutes. Batteries are reported as being either above or below 50 percent in capacity. Batteries below 50 percent in capacity are also referred to as being low (see "Fault-Tolerance for Write-Back Caching," page 2-8).

This 4-minute polling continues for up to 10 hours—the maximum time it should take to recharge the batteries. If the batteries have not been charged sufficiently after 10 hours, the controller declares them to be failed.

When charging a battery, write-back caching will be allowed as long as a previous down time has not drained more than 50 percent of a battery's capacity. When a battery is operating below 50 percent capacity, the battery is considered to be low and write-back caching is disabled.

Caution   It is recommended that you replace the ECB every two years to prevent battery failure and possible data loss.

Note   If an uninterruptible power supply (UPS) is used for backup power, the controller does not check the battery. See *HSZ70 Array Controller HSOF V7.3 CLI Reference Manual* for information about the *CACHE_UPS* and *NOCACHE_UPS* switches for the SET *controller* command.

C H A P T E R 2

Planning a Subsystem

This chapter contains concepts that will help you plan your subsystem:

When you have conceptually planned the subsystem, go to Chapter 3 and follow the guidelines for planning the storage devices. After the planning of the subsystem is complete, go to Chapter 4 for a configuration flowchart and a sample configuration procedure to help you implement your planned configuration.

Terminology

When configuring the subsystem you will encounter the following terms and concepts that you must understand:

■   Controller "A" and Controller "B"

■   "this" controller and "other" controller

Controller Designations A and B

Controllers and cache modules are designated either A or B depending on their location within the storage enclosure (shown in Figure 2–1).

Figure 2–1   Location of Controllers and Cache Modules

EMU	PVA
Controller A	
Controller B	
Cache module A	Cache module B
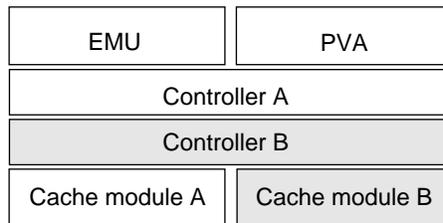
CXO6283A

Controller Designations "This" and "Other"

Some CLI commands use the terms "this" and "other" to identify one controller or the other in a dual-redundant pair. These designations are defined as follows:

■   "this" controller—the controller which is the focus of the CLI session. That is, the controller through which the CLI commands are being entered (may be controller A or B). The maintenance terminal is connected to the maintenance port of "this" controller.

■   "other" controller—the controller which is not the focus of the CLI session and through which CLI commands are not being entered. The maintenance terminal is not connected to the "other" controller.

The relationships are by default, a relative one, and are illustrated in Figure 2–2.

Figure 2–2   "This" Controller and "Other" Controller

OTHER_CONTROLLER

THIS_CONROLLER

CXO5716B
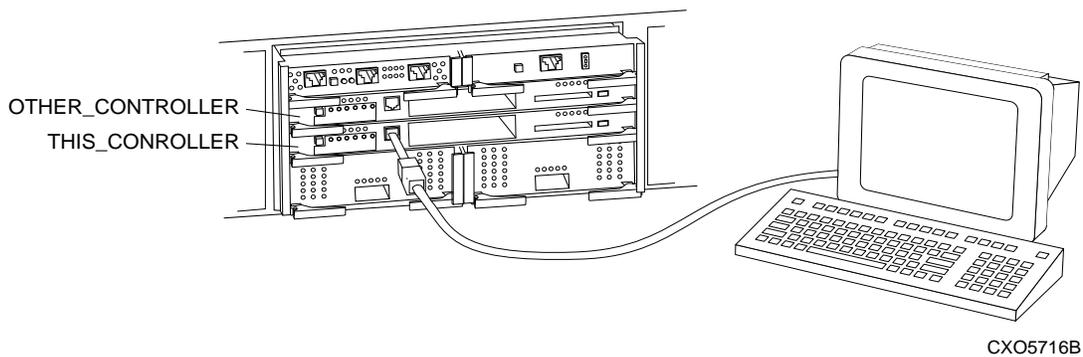
Selecting a Failover Mode

Failover is a way to keep the storage array available to the host in the event of a controller failure by allowing the surviving controller to take control of the entire subsystem. There are two modes of subsystem failover to choose from:

■   Transparent failover—occurs automatically, without intervention by the host.

■   Multiple-bus failover—occurs with assistance from the host which sends commands to the surviving controller.

Transparent Failover Mode

Transparent failover mode is used in a dual-redundant configuration in which both controllers are connected to the same host and device buses. Use transparent failover mode if you want both controllers to service the entire subsystem. Because both controllers service the same storage units, either controller can continue to service all of the units if its companion controller fails.

You can specify which targets are normally serviced by (assigned) a specific controller of the dual-redundant pair. This process is called preferring (or preferment). In transparent failover mode, targets can be preferred (assigned) to one controller or the other by the *PREFERRED_ID* switch of the SET *this_controller* (SET *other_controller*) CLI command. See the

HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual for detailed information about preferring units.

Keep the following tips in mind when planning for a transparent failover configuration:

■   Set each controller for transparent failover before configuring devices. When this step is done first, all devices, storagesets, and units added to one controller's configuration are automatically added to the other controller's configuration.

■   If you decide to configure the subsystem storage devices before setting the controllers to transparent failover, ensure you know which controller has the good configuration information before entering the CLI command: SET FAILOVER COPY =.

■   For better subsystem performance, balance your assignment of target ID numbers across your dual-redundant controller pair. For example, if you are presenting four targets to the host, prefer two targets to "this" controller and two targets to "other" controller.

Multiple-Bus Failover Mode

Multiple-bus (host-assisted) failover is a dual-redundant controller configuration in which each host has two paths (SCSI buses) to the array controller subsystem. The host(s) have the capability to move LUNs from one path to the other. With this capability, if one path fails, the host(s) can move all storage to the surviving path. Because both controllers service the same storage units, either controller can continue to service all of the units if the companion controller fails.

Keep the following tips in mind when planning for configuring the controllers for multiple-bus failover:

■   Multiple-bus failover can compensate for a failed controller, SCSI bus, or host bus adapter.

■   The host(s) can re-distribute the I/O load between the controllers.

■   The host(s) must have operating system software that supports multiple-bus failover mode (see *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for detailed information on the use of the CLI command: SET MULTIBUS_FAILOVER *copy=*).

■   Each host must connect to the controllers through two host bus adapters and two SCSI buses.

■   Multiple-bus failover does not support storage device partitioning (you must delete any existing partitions before enabling multiple-bus failover, and you cannot create partitions once multiple-bus failover is in effect).

■   Pass-through devices (tape and CD-based storage devices) cannot be supported by an HSZ70 dual-redundant pair operating in multiple-bus failover mode. This restriction is inherent in the architecture of the pass-through concept and the mechanisms by which the host operating system is aware of device location.

Customers using pass-through devices should segment their storage configuration by:

–   Placing the pass-through devices on controllers operating in transparent failover mode.

–   Placing the disk devices on controllers operating in multiple-bus failover mode.

Configuration Rules

Before configuring the controller, review these configuration rules and ensure your configuration meets the requirements and conditions.

■   Maximum 64 assignable, host-visible LUNs
(63 assignable when using StorageWorks Command Console)

■   Maximum 512 GB LUN capacity

■   Maximum 72 physical devices

■   Maximum 20 RAID-5 storagesets

■   Maximum 30 RAID-5 and RAID-1 storagesets for dual controller configurations.

■   Maximum 20 RAID-5 and RAID-1 storagesets for single controller configurations.

■   Maximum 45 RAID-5, RAID-1, and RAID-0 storagesets

■   Maximum 8 partitions per storageset or individual disk

■   Maximum 6 members per mirrorset

■   Maximum 14 members per RAIDset or stripeset

■   Maximum 32 physical device members total for a unit

■   Maximum 1 external tape device per device port. If you have an external tape drive on a port, you cannot configure any other devices (disks or tapes) on that port.

■   Maximum 1 internal (within an SBB) tape per device port. You can configure disks in the remaining slots on the port.

Selecting a Cache Mode

Before selecting a cache mode you should understand the caching techniques supported by the cache module. The cache module supports the following caching techniques to increase the performance of the subsystem read and write operations:

■   Read caching

■   Read-ahead caching

■   Write-through caching

■   Write-back caching

Read Caching

This caching technique decreases the subsystem response time to a read request by allowing the controller to satisfy the request from the cache memory rather than from the disk drives.

When the controller receives a read request from the host, it reads the data from the disk drives, delivers it to the host, and also stores it in the cache memory. If the host requests the same data again, the controller can satisfy the read request from the cached data rather than re-reading it from the disk drives.

Read caching can decrease the subsystem response time to many of the host read requests. If the host requests some or all of the cached data, the controller satisfies the request from its cache module rather than from the disk drives. By default, read caching is enabled for all storage units.

See the SET *unit* command *MAXIMUM_CACHED_TRANSFER* switch in the *HSZ70 Array Controller HSOF V7.3 CLI Reference Manual* for more detail.

Read-Ahead Caching

Read-ahead caching begins once the controller has already processed a read request and it receives a sequential read request from the host. If

the controller does not find the data in the cache memory, it reads the data from the disks and sends it to the cache memory.
The controller then anticipates subsequent read requests and begins to prefetch the next blocks of data from the disks as it sends the requested read data to the host. This is a parallel action. The controller notifies the host of the read completion, and subsequent sequential read requests are satisfied through the cache memory. By default, read-ahead caching is enabled for all disk units.

Write-Through Caching

This caching technique also decreases the subsystem response time to a read request by allowing the controller to satisfy the request from the cache memory rather than from the disk drives.

When the controller receives a write request from the host, it stores the data in its cache memory, writes the data to the disk drives, then notifies the host when the write operation is complete. This process is called write-through caching because the data actually passes through—and is stored in—the cache memory on its way to the disk drives.

If the host requests the recently written data, the controller satisfies the read request from its cache memory rather than from the disk drives. This caching technique is called write-through caching because the write data passes through—and is stored in—the cache memory on its way to the target disk drives.

If read caching is enabled for a storage unit, write-through caching is also enabled. Also, because both caching techniques enhance the controller's read performance, if you disable read caching, write-through caching is automatically disabled.

By default, read caching—and therefore write-through caching—is enabled for all storage units.

Write-Back Caching

This caching technique decreases the subsystem response time to write requests by allowing the controller to declare the write operation complete as soon as the data reaches its cache memory. The controller performs the slower operation of writing the data to the disk drives at a later time.

If the mirrorset is a disaster-tolerant mirrorset, then write-back caching is not enabled, nor can it be enabled.

By default, write-back caching is disabled for all storagesets. In either case, the controller will not provide write-back caching to a unit unless you ensure that the cache memory is non-volatile as described below.

Fault-Tolerance for Write-Back Caching

The cache module supports the following features to protect the availability of its unwritten (write-back) data:

■ Non-volatile memory (required for write-back caching)

■ Mirrored caching (optional)

■ Dynamic caching techniques (automatic)

Non-Volatile Memory

The controller cannot provide write-back caching to a unit unless its cache memory is non-volatile. In other words, you must provide a back-up power source to the cache module to preserve the unwritten cache data in the event of a power failure. If the cache memory were volatile—that is, if it weren't backed up with backup power—the unwritten cache data would be lost during a power failure.

By default, the controller expects to use an ECB as the cache module back-up power source. See the *HSZ70 Array Controller HSOF Version 7.3 Service Manual* service manual for more information about the ECB. If the subsystem is backed up by a UPS, then a switch must be set in the SET *this_controller* (SET *other_controller)* command.

See the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details about using theCLI Command: SET *this_controller CACHE_UPS.*

However, if your subsystem is backed up by a UPS, you can tell the controller to use the UPS as the backup power source with the CLI Command: SET *this_controller CACHE_UPS*. See the *HSZ70 Array Controller HSOF V7.3 CLI Reference Manual* for instructions on using this command.

Note   The controller executes multiple write operations to satisfy a single write request for a RAIDset or mirrorset. For this reason, a RAIDset or mirrorset requires non-volatile cache memory to ensure data redundancy until the write request is satisfied.

Regardless of the back-up power source you choose, the cache-memory power LED flashes about once every three seconds to indicate the cache module's memory array is receiving power from its primary power source.

Mirrored Caching

To further ensure the availability of unwritten cache data, you can use a portion of each cache module memory to mirror the other cache module's write-back data in a dual-redundant configuration.

Figure 2–3 shows the principle of mirrored caching: half of cache A mirrors cache B write-back data and vice versa. This arrangement ensures that the write-back data will be preserved if a cache module or any of its components fail.

Note   When your controllers are configured to use mirrored write-back cache, the cache capacity is half of the total amount of cache in the configuration. If each cache module has 64 MB of cache for a total of 128 MB of cache in the configuration, the cache capacity is 64 MB.

Figure 2–3   Mirrored Caching

Cache module A   Cache module B

| A cache | B cache |
| Copy of B cache | Copy of A cache |
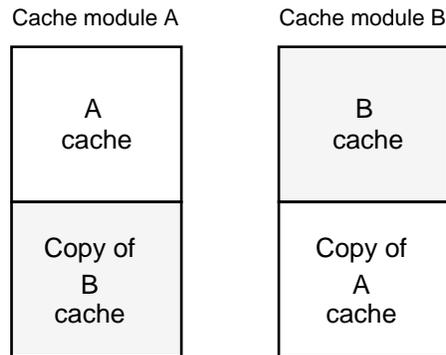
CXO5729A

Before configuring dual-redundant controllers and enabling mirrored write-back cache, ensure the following conditions are met:

■   Both controllers have the same size cache, 64 MB or 128 MB.

■   Diagnostics indicates that both cache modules are good.

■   Both cache modules have a battery present (unless if you have enabled the *CACHE_UPS* switch). A battery does not have to be present for either cache if you enable the *CACHE_UPS* switch.

■   No unit errors are outstanding, for example, lost data or data that cannot be written to devices.

■   Both controllers are started and configured in failover mode.

For important considerations when replacing or upgrading SIMMs in a mirrored cache configuration, see the *HSZ70 Array Controller HSOF Version 7.3 Service Manual*

Dynamic Caching Techniques

If the controller detects a full or partial failure of its cache module or ECB, it automatically reacts to preserve the cached write-back data. Then, depending upon the severity of the failure, the controller chooses an interim caching technique—also called the cache policy—which it uses until you repair or replace the cache module or its ECB.

Cache Policies as a Result of Failures

If the controller detects a full or partial failure of its cache module or ECB, it automatically reacts to preserve the unwritten data in its cache module. Depending upon the severity of the failure, the controller chooses an interim caching technique—also called the cache policy—which it uses until you repair or replace the cache module.

Table 2–1 shows the consequences of a full or partial failure of cache module A in a dual-redundant configuration. The consequences shown in this table are reciprocal for a failure of cache module B.

Table 2–1   Cache Policies and Cache Module Status

Cache Module Status		Cache Policy	
Cache A	**Cache B**	**Unmirrored Cache**	**Mirrored Cache**
Good	Good	**Data loss:** No. **Cache policy:** Both controllers support write-back caching. **Failover:** No.	**Data loss:** No. **Cache policy:** Both controllers support write-back caching. **Failover:** No.
Multibit cache memory failure	Good	**Data loss:** Forced error and loss of write-back data for which multibit error occurred. Controller A detects and reports the lost blocks. **Cache policy:** Both controllers support write-back caching. **Failover:** No.	**Data loss:** No. Controller A recovers its lost write-back data from the mirrored copy on cache B. **Cache policy:** Both controllers support write-back caching. **Failover:** No.

Table 2– 1   Cache Policies and Cache Module Status (Continued)

Cache Module Status		Cache Policy	
Cache A	**Cache B**	**Unmirrored Cache**	**Mirrored Cache**
SIMM or cache memory controller failure	Good	**Data loss:** Loss of write-back data that wasn't written to media when failure occurred. **Cache policy:** Controller A supports write-through caching only; controller B supports write-back caching. **Failover:** In transparent failover, all units failover to controller B. In multiple-bus failover, only those units that use write-back caching, such as RAIDsets and mirrorsets, failover to controller B. All units with lost data become inoperative until you clear them with the CLEAR LOST_DATA command. Units that didn't lose data operate normally on controller B. In single configurations, RAIDsets, mirrorsets, and all units with lost data become inoperative. Although you can clear the lost data errors on some units, RAIDsets and mirrorsets remain inoperative until you repair or replace the non-volatile memory on cache A.	**Data loss: No.** Controller A recovers all of its write-back data from the mirrored copy on cache B. **Cache policy:** Controller A supports write-through caching only; controller B supports write-back caching. **Failover:** In transparent failover, all units failover to controller B and operate normally. In multiple-bus failover, only those units that use write-back caching, such as RAIDsets and mirrorsets, failover to controller B.
Cache board failure	Good	Same as for SIMM failure.	**Data loss: No.** Controller A recovers all of its write-back data from the mirrored copy on cache B. **Cache policy:** Both controllers support write-through caching only. Controller B cannot execute mirrored writes because cache module A cannot mirror controller B's unwritten data. **Failover:** No.

Table 2– 2 shows the consequences of a full or partial failure of the ECB for cache module A (dual-redundant configuration). The consequences shown in this table are reciprocal for a failure of the ECB for cache module B.

Table 2– 2   Cache Policies and ECB Status

ECB Status		Cache Policy	
ECB Cache A	**ECB Cache B**	**Unmirrored Cache**	**Mirrored Cache**
Good	Good	**Data loss:** No. **Cache policy:** Both controllers continue to support write-back caching. **Failover:** No.	**Data loss:** No. **Cache policy:** Both controllers continue to support write-back caching. **Failover:** No.
Low	Good	**Data loss:** No. **Cache policy:** Controller A supports write-through caching only; controller B supports write-back caching. **Failover:** In transparent failover, all units failover to controller B and operate normally. In multiple-bus failover, only those units that use write-back caching, such as RAIDsets and mirrorsets, failover to controller B. In single configurations, the controller only provides write-through caching to its units.	**Data loss:** No. **Cache policy:** Both controllers continue to support write-back caching. **Failover:** No.

Table 2– 2   Cache Policies and ECB Status (Continued)

ECB Status		Cache Policy	
ECB Cache A	**ECB Cache B**	**Unmirrored Cache**	**Mirrored Cache**
Failed	Good	**Data loss:** No. **Cache policy:** Controller A supports write-through caching only; controller B supports write-back caching. **Failover:** In transparent failover, all units failover to controller B and operate normally. In multiple-bus failover, only those units that use write-back caching, such as RAIDsets and mirrorsets, failover to controller B. In single configurations, RAIDsets and mirrorsets become inoperative.	**Data loss:** No. **Cache policy:** Both controllers continue to support write-back caching. **Failover:** No.
Low	Low	**Data loss:** No. **Cache policy:** Both controllers support write-through caching only. **Failover:** No.	**Data loss:** No. **Cache policy:** Both controllers support write-through caching only. **Failover:** No.
Failed	Low	**Data loss:** No. **Cache policy:** Both controllers support write-through caching only. **Failover:** In transparent failover, all units failover to controller B and operate normally. In multiple-bus failover, only those units that use write-back caching, such as RAIDsets and mirrorsets, failover to controller B. In single configurations, RAIDsets and mirrorsets become inoperative.	**Data loss:** No. **Cache policy:** Both controllers support write-through caching only. **Failover: No.**

Table 2– 2   Cache Policies and ECB Status (Continued)

ECB Status		Cache Policy	
ECB Cache A	ECB Cache B	Unmirrored Cache	Mirrored Cache
Failed	Failed	**Data loss:** No.	**Data loss:** No.
		Cache policy: Both controllers support write-through caching only.	**Cache policy:** Both controllers support write-through caching only.
		Failover: No. **RAIDsets and mirrorsets** become inoperative. Other units that use write-back caching operate with write-through caching only.	**Failover:** No. **RAIDsets and mirrorsets** become inoperative. Other **u**nits that use write-back caching operate with write-through caching only.

Assigning Targets

A target is a SCSI ID on a SCSI bus or buses. The logical units (LUNs) that the controllers present to the host(s) are each bound to a specific host target ID. The user must inform the host (configure) the specific targets that are being used for this particular subsystem.

The range of host SCSI target ID numbers available is 0 through 15 (excluding any IDs already taken by other devices on the host SCSI bus). Usually storage targets have SCSI IDs between 0—5 and 8—15 because of the following considerations:

- It is good practice to have a maximum of two hosts per SCSI bus.
- It is also good practice to maintain maximum throughput between host(s) and the storage array by not having any other device(s) on the SCSI bus.
- It is highly recommended to assign the host adapters to the highest SCSI arbitration IDs, which are ID 6 and ID 7.

Assigning the host SCSI target ID is a function of the user entering the CLI command:
SET *this_controller ID=*

> **Note**   This command **must** be entered **before** entering any unit
> numbers with the ADD UNIT *unit-number container-name* command.
> The host will not recognize any unit number that contains a host SCSI
> Target ID that has not already been entered into the system.

Figure 2–4 shows the SCSI IDs of a typical, normally configured
transparent failover configuration. Figure 2–5 shows the SCSI IDs of a
typical, normally configured multiple-bus failover configuration.

Figure 2–4   Transparent Failover Mode
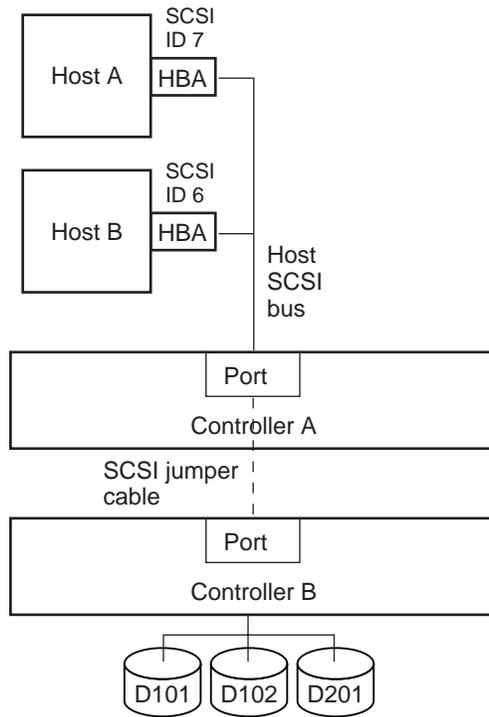
CXO6678A

Figure 2–5   Multiple-Bus Failover Mode
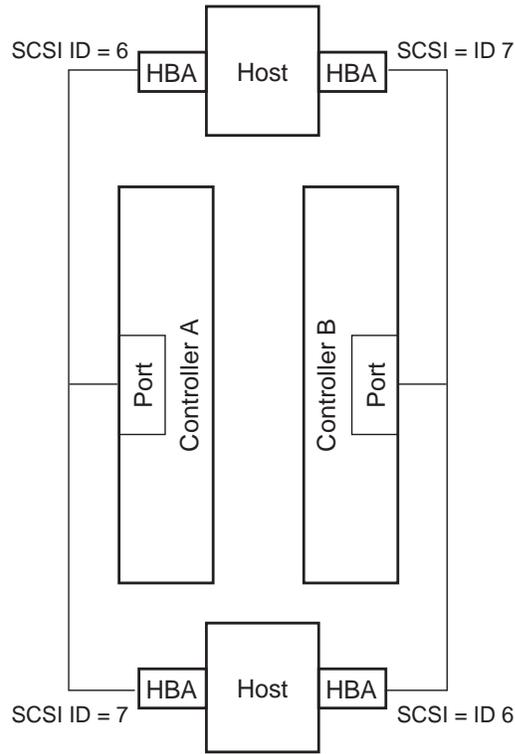
CXO6679B

Assigning Logical Unit Numbers

Every container (storageset, partition, or JBOD disk) needs a logical unit number (LUN) in order to communicate to the host(s). Each LUN on the host side contains the following:

■   A letter that indicates the kind of devices in the storage unit:

– D= disk drives.

– P= passthrough devices such as tape drives, loaders, and libraries).

> **Note**   If the CFMENU utility is used to configure the
> storagesets and devices, it automatically supplies a device
> letter.

■ First Number—indicates the valid one-of-eight SCSI target IDs.
    The number selected must be in the range of those numbers already
    placed into the system (see "Assigning Targets," page 2-15).

> **Note**   By carefully choosing the first number, you can
> establish preferred paths for all of your storage units in a dual-
> redundant configuration.

■ Second number—Always zero.
■ Third number—Identifies the logical unit number (LUN) for the
    device or storage unit (can be a number from 0—7).

Omit any leading zeroes for storage units associated with the controller
SCSI target ID of zero. For example, use D2 instead of D002 for a
container that is accessed through LUN 2 of the controller's SCSI
target ID 0. Table 2–3 shows additional unit numbering examples.

> **Note**   The host communicates with a logical unit based on its LUN
> address. The controller communicates with a storage device based on
> its PTL address.

Table 2–3   Unit Numbering Examples

Unit Number	Device Type	Target ID Number	LUN
D401	disk	4	1
D1207	disk	12	7
P5	tape	0	5

The Unit Number (LUN) is placed into the system with the ADD UNIT *unit-number container-name command.* This command assigns a Unit Number to the container name established with the ADD DISK command.

For both of these commands see the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details.

Command Console LUN

When the controller is unconfigured and the StorageWorks command console (SWCC) is to be used to configure the controller, a virtual LUN (Command Console LUN; CCL) needs to be established. Setting a CCL allows the SWCC to communicate with the controller (the default is no CCL).

Establish a CCL by using the SET *this_controller* (SET *other_controller*) *COMMAND_CONSOLE_LUN* CLI command. See the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details.

Host Modes

Different operating systems handle SCSI protocols and commands differently. Due to this situation, each target is assigned a unique host function mode. The operating-system unique host function mode enables the target to communicate in a fashion compatible with the host operating system. The default host function mode is "A".

To view the host function mode settings on the controller, use the following syntax:

SHOW THIS_CONTROLLER FULL

SHOW OTHER_CONTROLLER FULL

You need only set the host mode if a target needs to communicate to a host operating system that is different from the default. See the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details on the host function mode.

Restricting Host Access

In a subsystem that is connected to more than one host, is possible to reserve certain units for the exclusive use by a specific host.
The method that is used to restrict access is by enabling/disabling the access path of a selected host.

In the SET *unit-number* CLI command, the *ACCESS_ID=* switch allows a SCSI ID number to be placed along with the unit number assigned controlling the access to a particular host.

See the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details.

C H A P T E R 3

Planning Storage

This chapter provides information to help you plan the storage configuration of your subsystem. Use the procedure found in this section to plan and configure the various types of storage containers needed.

Containers are individual disk drives (JBOD), storageset types, and/or partitioned drives and storagesets. Use the references in each step to locate details about specific commands and concepts. Appendix B contains blank templates for keeping track of the containers being configured.

1. Familiarize yourself with the physical layout of the devices. See Chapter 1 for "Physically Connecting the Host to the Storage Array" and "Logically Connecting the Storage Array to the Host".

2. Determine your storage requirements. Use the questions in "Determining Storage Requirements" page 3 to help you.

3. Choose the type of storage container(s) you need to use in your subsystem. See "Choosing a Container Type" page 3 for a comparison and description of each type of storageset.

4. Create a storageset profile (described in "Creating a Storageset Profile" page 3). Fill out the storageset profile while you read the sections that pertain to your chosen storage type:

   ■ "Stripeset Planning Considerations," page 3-6

   ■ "Mirrorset Planning Considerations," page 3-9

   ■ "Partition Planning Considerations," page 3-17

   ■ "Striped Mirrorset Planning Considerations," page 3-16

5. Decide on which switches you will need for your subsystem. Device switches apply to all devices, including those configured as single disk units (JBOD). General information on switches are detailed in "Choosing Switches for Storagesets and Devices," page 3-19

■   Determine what unit switches you want for your units ("Device Switches," page 3-23)

■   Determine what initialization switches you want for your planned storage containers ("Initialize Command Switches," page 3-25).

6.   Create a storage map ("Storage Maps," page 3-35).

7.   Configure the storage you have now planned using one of the following methods:

■   Use the StorageWorks Command Console (SWCC) graphical user interface (GUI). See the SWCC documentation for details regarding the use of the GUI to configure your storage.

■   Use Command Line Interpreter (CLI) commands by way of a terminal or PC connected to the maintenance port of the controller. This method allows you flexibility in defining and naming your storage containers. Chapter 4 contains the procedures for configuring your storage with the CLI. The *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* contains the details of each CLI command.

Determining Storage Requirements

Planning storagesets for your system involves determining what your storage requirements are. Here are a few of the questions you should ask yourself of the subsystem usage:

■   What applications or user groups will access the subsystem? How much capacity do they need?

■   What are the I/O requirements? If an application is data-transfer intensive, what is the required transfer rate? If it is I/O-request intensive, what is the required response time? What is the read/write ratio for a typical request?

■   Are most I/O requests directed to a small percentage of the disk drives? Do you want to keep it that way or balance the I/O load?

■   Do you store mission-critical data? Is availability the highest priority or would standard backup procedures suffice?

Choosing a Container Type

Different applications may have different storage requirements, so you will probably want to configure more than one kind of container in your subsystem.

In choosing a container, you have the following choices:

■ Independent disks (JBODs)

■ Storagesets

■ Partitioned Storagesets

■ Partitioned Disks

The storagesets described in this book implement RAID (Redundant Array of Independent Disks) technology. Consequently, they all share one important feature: each storageset, whether it contains two disk drives or ten, looks like one large, virtual disk drive to the host.

Table 3– 1 compares the different kinds of containers to help you determine which ones satisfy your requirements.

Table 3– 1   A Comparison of Container Types

Storageset	Relative Availability	Request Rate (Read/Write) I/O per second	Transfer Rate (Read/Write) MB per second	Applications
Array of independent disk drives (JBOD)	Proportionate to number of disk drives	Identical to single disk drive	Identical to single disk drive	
Stripeset (RAID 0)	Proportionate to number of disk drives; worse than single disk drive	Excellent if used with large chunk size	Excellent if used with small chunk size	High performance for non-critical data
Mirrorset (RAID1)	Excellent	Good/Fair	Good/Fair	System drives; critical files
RAIDset (RAID 3/5)	Excellent	Excellent/Fair	Good/Poor	High request rates, read-intensive, data lookup
Striped Mirrorset (RAID 0+1)	Excellent	Excellent if used with large chunk size	Excellent if used with small chunk size	Any critical response-time application

For a comprehensive discussion of RAID, refer to *The RAIDBOOK—A Source Book for Disk Array Technology.*

Creating a Storageset Profile

Creating a profile for your storagesets, partitions, and devices can help simplify the configuration process. Filling out a storageset profile helps you to choose the storagesets that best suit your needs and to make informed decisions about the switches that you can enable for each storageset or storage device that you configure in your subsystem.

Familiarize yourself with the kinds of information contained in the example storageset profile, as shown in Figure 3–1.

Appendix B contains blank profiles that you can copy and use to record the details for your storagesets. Use the information in this chapter to help you make decisions when creating storageset profiles.

Figure 3–1   An Example Storageset Profile

Type of Storageset:

_____ Mirrorset	__X_ RAIDset	_____ Stripeset	_____ Striped Mirrorset	____ JBOD

Storageset Name...........*R1*

Disk Drives....................*D10300, D20300, D30300, D40300, D50300, D60300*

Unit Number.................*D101*

Partitions:

Unit #	Unit #	Unit #	Unit #	Unit #	Unit #	Unit #	Unit #
%	%	%	%	%	%	%	%

RAIDset Switches:

Reconstruction Policy	Reduced Membership	Replacement Policy
_X_Normal (default) ___Fast	_X_ No (default) ___Yes, missing:	_X_Best performance (default) ___Best fit ___None

Mirrorset Switches:

Replacement Policy	Copy Policy	Read Source
___Best performance (default) ___Best fit ___None	___Normal (default) ___Fast	___Least busy (default) ___Round robin ___Disk drive:

Initialize Switches:

Chunk size	Save Configuration	Metadata
_X_ Automatic (default) ___ 64 blocks ___ 128 blocks ___ 256 blocks ___ Other:	___No (default) _X_Yes	_X_Destroy (default) ___Retain

Unit Switches:

Read Cache	Write Cache	Maximum Cache Transfer
_X_ Yes (default) ___ No	___Yes (default) _X_No	_X_32 blocks (default) ___Other:
Write-Back Cache	**Write Protection**	**Availability**
___Yes (default) _X_No	_X_No (default) Yes	_X_Run (default) ___NoRun

Host Access Enabled: _____

Stripeset Planning Considerations

Stripesets (RAID 0) enhance I/O performance by spreading the data across multiple disk drives. Each I/O request is broken into small segments called "chunks." These chunks are then "striped" across the disk drives in the storageset, thereby allowing several disk drives to participate in one I/O request to handle several I/O requests simultaneously.

For example, in a three-member stripeset that contains disk drives 10000, 20000, and 30000, the first chunk of an I/O request is written to 10000, the second to 20000, the third to 30000, the fourth to 10000, and so forth until all of the data has been written to the drives (Figure 3–2).

Figure 3–2   Example 3-Member Stripeset
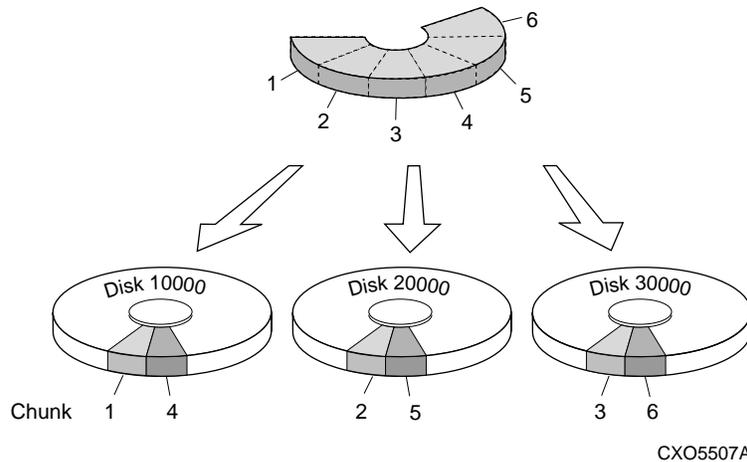
CXO5507A

The relationship between the chunk size and the average request size determines if striping maximizes the request rate or the data-transfer rate. You can set the chunk size or let the controller set it automatically. See "Chunk Size" page 3 , for information about setting the chunk size. Figure 3–3 shows another example of a three-member RAID 0 Stripeset.

An major benefit of striping is that it balances the I/O load across all of the disk drives in the storageset. This can increase the subsystem performance by eliminating the hot spots (high localities of reference), that occur when frequently accessed data becomes concentrated on a single disk drive.

Figure 3–3   Stripeset Example 2

CXO4592B
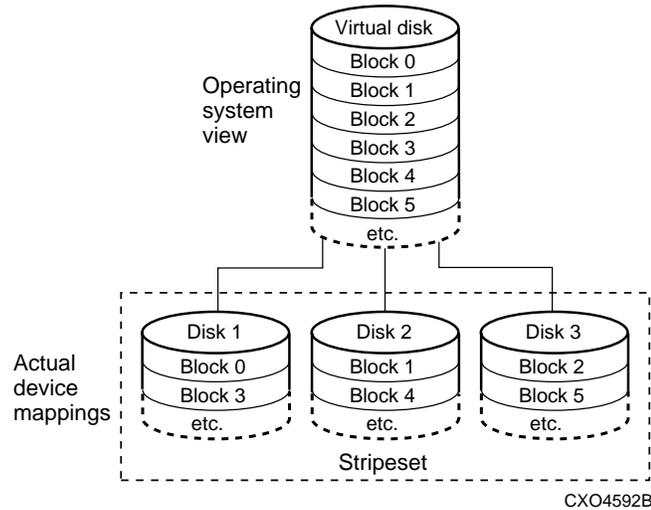
Keep the following points in mind as you plan your stripesets:

■   A controller can support the following storageset maximums:

RAIDSET Type	Limit
RAID 5	20
RAID 5+RAID 1	30
RAID 5 +RAID 1 +RAID 0	45

■   Reporting methods and size limitations prevent certain operating
      systems from working with large stripesets. See the *StorageWorks
      Array Controller HSZ70 Array Controller Operating Software
      HSOF Version 7.3 Release Notes* or the *Getting Started Guide* that
      came with your platform kit for details about these restrictions.

■   A storageset should only contain disk drives of the same capacity.
      The controller limits the capacity of each member to the capacity of
      the smallest member in the storageset (base member size) when the
      storageset is initialized. Thus, if you combine 2 GB disk drives with
      1 GB disk drives in the same storageset, you will waste 1 GB of
      capacity on each 2 GB member.

If you need high performance and high availability, consider using a RAIDset, striped mirrorset, or a host-based shadow of a stripeset.

■ Striping does not protect against data loss. In fact, because the failure of one member is equivalent to the failure of the entire stripeset, the likelihood of losing data is higher for a stripeset than for a single disk drive.

For example, if the mean time between failures (MTBF) for a single disk is l hour, then the MTBF for a stripeset that comprises N such disks is l/N hours. As another example, if the MTBF of a a single disk is 150,000 hours (about 17 years), a stripeset comprising four of these disks would only have an MTBF of slightly more than 4 years.

For this reason, you should avoid using a stripeset to store critical data. Stripesets are more suitable for storing data that can be reproduced easily or whose loss does not prevent the system from supporting its critical mission.

■ Evenly distribute the members across the device ports to balance load and provide multiple paths as shown in the Figure 3–4.

■ Stripesets for HSOF V7.3 contain between two and 14 members.

■ Stripesets are well-suited for the following applications:

– Storing program image libraries or run-time libraries for rapid loading

– Storing large tables or other structures of read-only data for rapid application access

– Collecting data from external sources at very high data transfer rates

■ Stripesets are not well-suited for the following applications:

– A storage solution for data that cannot be easily reproduced or for data that must be available for system operation

– Applications that make requests for small amounts of sequentially located data

– Applications that make synchronous random requests for small amounts of data

Figure 3–4   Distribute Members Across Device Ports
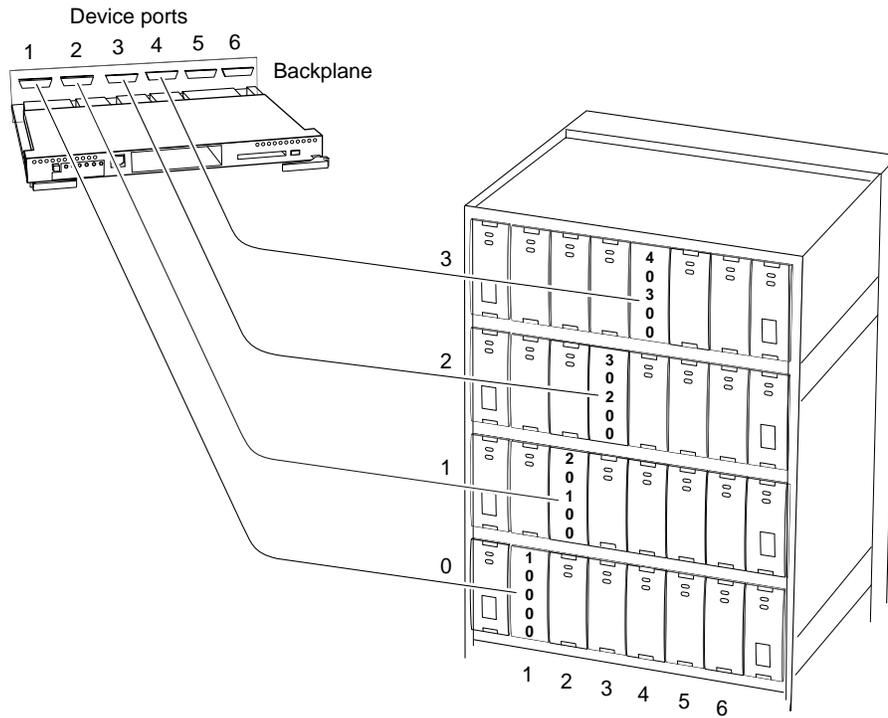
CXO6235A

By spreading the traffic evenly across the buses, you will ensure that no bus handles the majority of data to the storageset.

Mirrorset Planning Considerations

Mirrorsets (RAID 1) use redundancy to ensure availability, as illustrated in Figure 3–5. For each primary disk drive, there is at least one mirror disk drive. Thus, if a primary disk drive fails, its mirror drive immediately provides an exact copy of the data. Figure 3–6 shows a second example of a Mirrorset.

Figure 3–5   Mirrorsets Maintain Two Copies of the Same Data
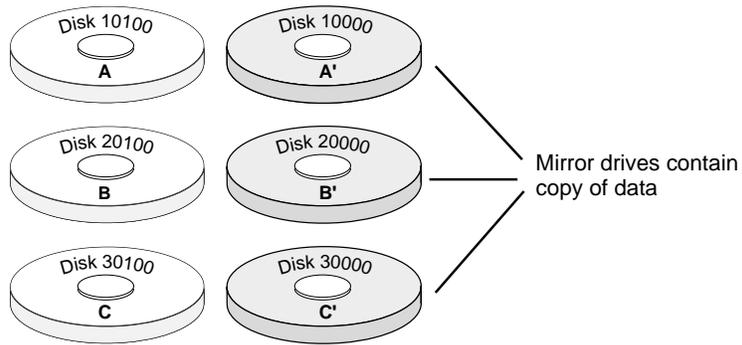
CXO5511A

Figure 3–6   Mirrorset Example 2
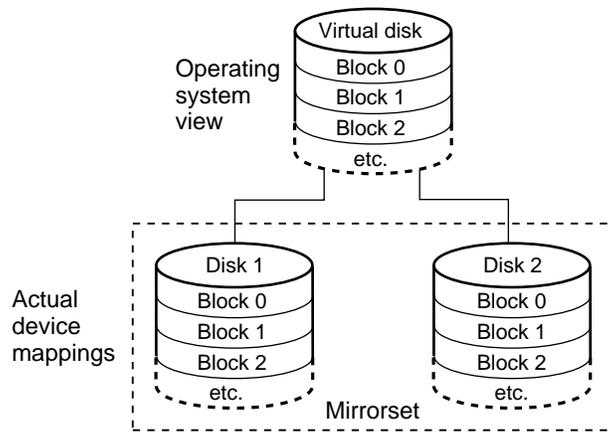
CXO4594B

Keep these points in mind as you plan your mirrorsets:

■   A controller can support the following storageset maximums:

RAIDSET Type	Limit
RAID 5	20
RAID 5+RAID 1	30
RAID 5 10+RAID 1 +RAID 0	45

■   Data availability with a mirrorset is excellent but comes with a
    higher cost—you need twice as many disk drives to satisfy a given
    capacity requirement. If availability is your top priority, consider
    using redundant power supplies and dual-redundant controllers.

■   You can configure up to 20 mirrorsets per controller or pair of dual-
    redundant controllers. Each mirrorset may contain up to 6
    members.

■   A write-back cache module is required for mirrorsets, but write-
    back cache need not be enabled for the mirrorset to function
    properly.

■   Both write-back cache modules must be the same size.

■   If you are using more than one mirrorset in your subsystem, you
    should put the first member of each mirrorset on different buses as
    shown in Figure 3–7. (The first member of a mirrorset is the first
    disk drive you add.)

    When a controller receives a request to read or write data to a
    mirrorset, it typically accesses the first member of the mirrorset. If
    you have several mirrorsets in your subsystem and their first
    members are on the same bus, that bus will be forced to handle the
    majority of traffic to your mirrorsets.

Figure 3–7   First Mirrorset Members Placed on Different Buses
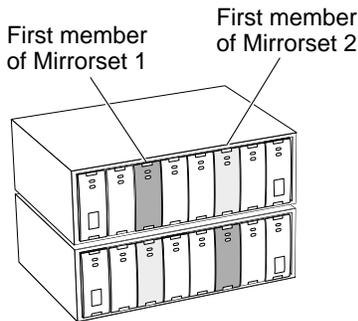
CXO5506A

To avoid an I/O bottleneck on one bus, you can simply put the first members on different buses. Additionally, you can set the read-source switch to round robin. See "Read Source" page 3 , for more information about this switch.

■ Place mirrorsets and RAIDsets on different ports to minimize risk in the event of a single port bus failure.

■ Mirrorset units are set to *NOWRITEBACK_CACHE* by default. To increase a unit's performance, switch to *WRITEBACK_CACHE*.

■ A storageset should only contain disk drives of the same capacity. The controller limits the capacity of each member to the capacity of the smallest member in the storageset. Thus, if you combine 2 GB disk drives with 1 GB disk drives in the same storageset, you waste 1 GB of capacity on each 2 GB member.

■ Evenly distribute the members across the device ports to balance load and provide multiple paths as shown in Figure 3–4 on page 3-9.

■ Mirrorsets **are** well-suited for the following:
   – Any data for which reliability requirements are extremely high
   – Data to which high-performance access is required
   – Applications for which cost is a secondary issue

■ Mirrorsets are **not** well-suited for the following applications:
   – Write-intensive applications
   – Applications for which cost is a primary issue

RAIDset Planning Considerations

RAIDsets (RAID 3/5) are enhanced stripesets—they use striping to increase I/O performance and distributed-parity data to ensure data availability. Figure 3–8 illustrates the concept of a three-member RAIDset. Figure 3–9 shows a second example of a RAIDset that uses five members.

Figure 3–8   Parity Ensures Availability; Striping for Performance
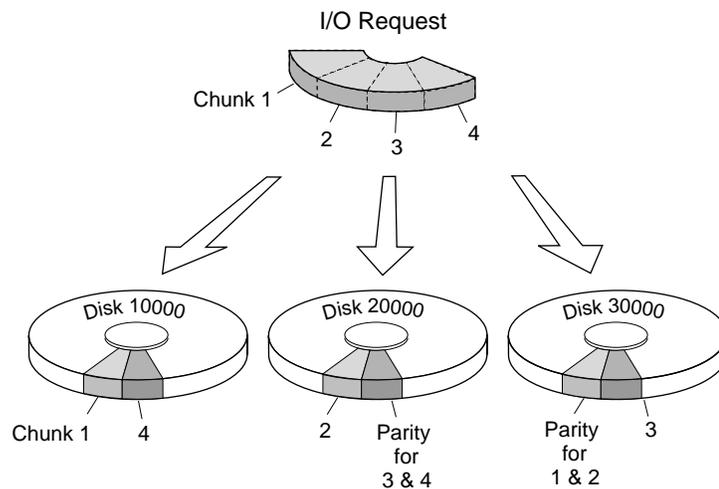
CXO5509A

RAIDsets are similar to stripesets in that the I/O requests are broken into smaller "chunks" and striped across the disk drives until the request is read or written. RAIDsets also create chunks of parity data and stripe them across the disk drives. This parity data is derived mathematically from the I/O data and enables the controller to reconstruct the I/O data if a disk drive fails. Thus, it becomes possible to lose a disk drive without losing access to the data it contained. Data could be lost, however, if a second disk drive fails before the controller replaces the first failed disk drive.

For example, in a three-member RAIDset that contains disk drives 10000, 20000, and 30000, the first chunk of an I/O request is written to 10000, the second to 20000, then parity is calculated and written to 30000; the third chunk is written to 30000, the fourth to 10000, and so on until all of the data is saved.

Figure 3–9   RAIDset Example 2
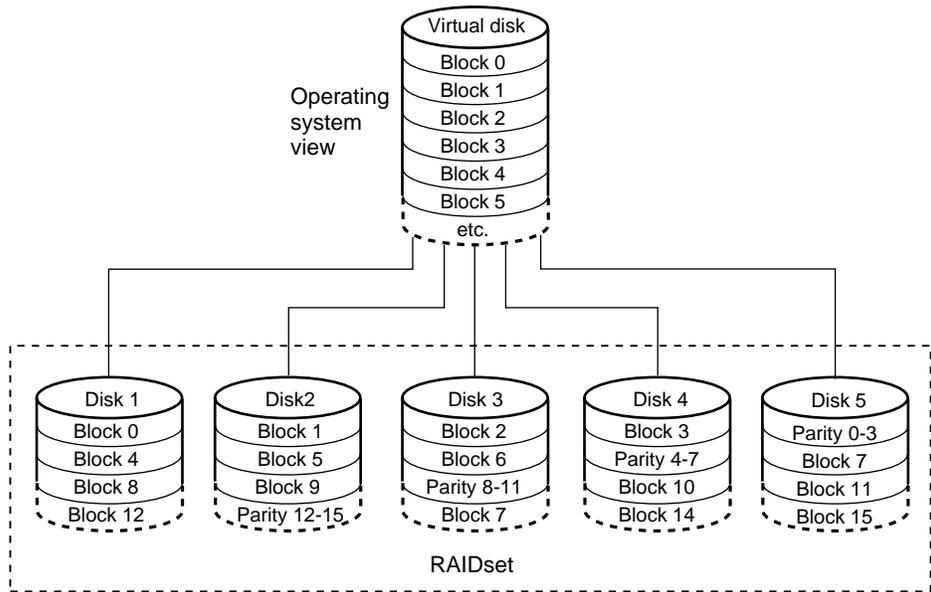
CXO6463A

The relationship between the chunk size and the average request size determines if striping maximizes the request rate or the data-transfer rates. You can set the chunk size or let the controller set it automatically. See "Chunk Size" page 3 , for information about setting the chunk size.

Keep these points in mind as you plan your RAIDsets:

■   A controller can support the following storageset maximums:

RAIDSET Type	Limit
RAID 5	20
RAID 5+RAID 1	30
RAID 5 +RAID 1 +RAID 0	45

■ Reporting methods and size limitations prevent certain operating systems from working with large RAIDsets. See the *StorageWorks Array Controller HSZ70 Array Controller Operating Software HSOF Version 7.3 Release Notes* or the *Getting Started Guide* that came with your platform kit for details about these restrictions.

■ A write-back cache module is required for RAIDsets, but write-back cache need not be enabled for the RAIDset to function properly.

■ Both write-back cache modules must be the same size.

■ A RAIDset must include at least 3 disk drives, but no more than 14.

■ Evenly distribute the members across the device ports to balance load and provide multiple paths as shown in Figure 3–4 on page 3-9.

■ A storageset should only contain disk drives of the same capacity. The controller limits the capacity of each member to the capacity of the smallest member in the storageset. Thus, if you combine 2 GB disk drives with 1 GB disk drives in the same storageset, you'll waste 1 GB of capacity on each 2 GB member.

■ RAIDset units are set to *NOWRITEBACK_CACHE* by default. To increase a unit's performance, switch to *WRITEBACK_CACHE*.

■ Place RAIDsets and mirrorsets on different ports to minimize risk in the event of a single port bus failure.

■ RAIDsets are particularly well-suited for the following:
   – Small to medium I/O requests
   – Applications requiring high availability
   – High read request rates
   – Inquiry-type transaction processing

■ RAIDsets are not particularly well-suited for the following:
   – Write-intensive applications
   – Applications that require high data transfer capacity
   – High-speed data collection
   – Database applications in which fields are continually updated
   – Transaction processing

Striped Mirrorset Planning Considerations

Striped mirrorsets (RAID 0+1) are a configuration of stripesets whose members are also mirrorsets. Consequently, this kind of storageset combines the performance of striping with the reliability of mirroring. The result is a storageset with very high I/O performance and high data availability (Figure 3–10). Figure 3–11 shows a second example of a striped mirrorset using six members.

Figure 3–10   Striping and Mirroring in the Same Storageset
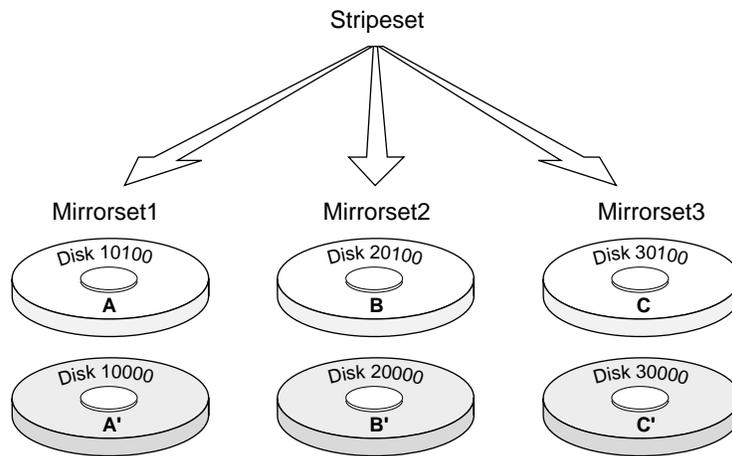
CXO5508A

The failure of a single disk drive has no effect on this storageset's ability to deliver data to the host and, under normal circumstances, it has very little effect on performance. Because striped mirrorsets do not require any more disk drives than mirrorsets, this storageset is an excellent choice for data that warrants mirroring.

Figure 3–11   Striped Mirrorset Example 2

CXO6462A
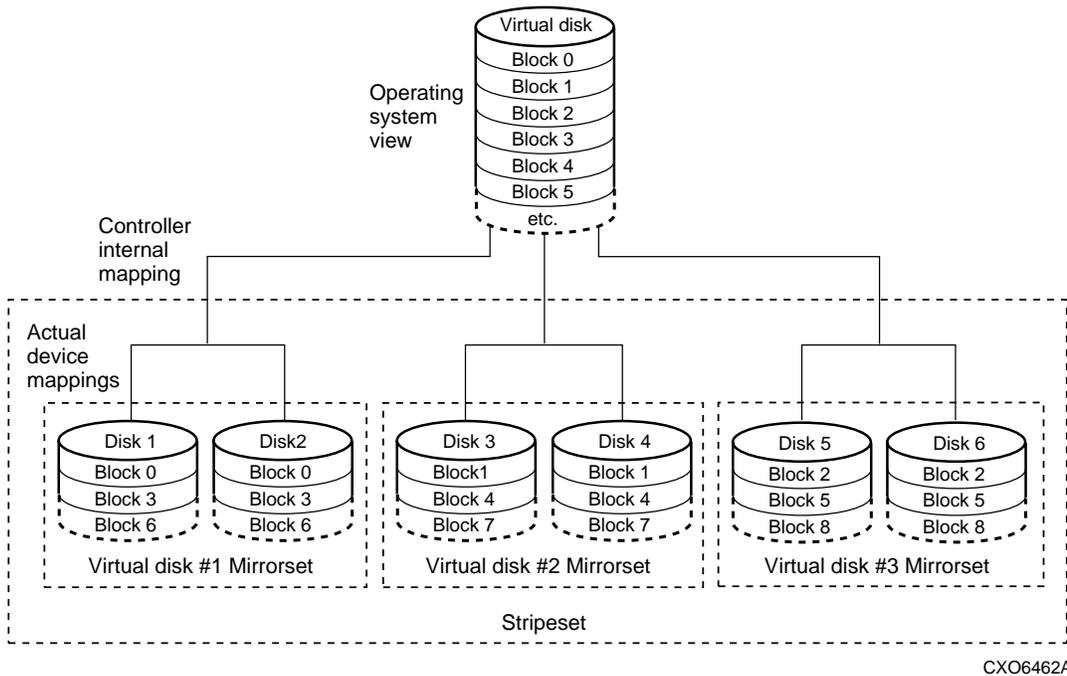
Plan the mirrorset members, then plan the stripeset that will contain them. Review the recommendations in "Stripeset Planning Considerations," page 3-6, and "Mirrorset Planning Considerations," page 3-9.

The following limitations exist when planning a striped mirrorset:

■   A maximum of 24 mirrorsets in a stripeset.

■   A maximum of 6 disks in each mirrorset.

■   A maximum of 48 disks in the entire striped mirrorset.

Partition Planning Considerations

Use partitions to divide a storageset or individual disk drive into smaller pieces, each of which can be presented to the host as its own

storage unit. Figure 3–12 shows the conceptual effects of partitioning a single-disk.

Figure 3–12   Partitioning a Single-Disk Unit
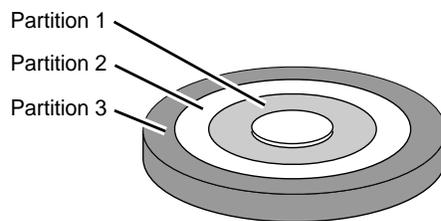
CXO-5316A-MC

You can create up to eight partitions per storageset (disk drive, RAIDset, mirrorset, stripeset, or striped mirrorset). Each partition has its own unit number so that the host can send I/O requests to the partition just as it would to any unpartitioned storageset or device. Partitions are separately addressable storage units, therefore, you can partition a single storageset to service more than one user group or application.

Defining a Partition

Partitions are expressed as a percentage of the storageset or single disk unit that contains them:

■   Mirrorsets and single disk units—the controller allocates the largest whole number of blocks that are equal to or less than the percentage you specify.

■   Stripesets—the stripe size = chunk size x number of members.

■   RAIDsets, the stripe size = chunk size x (number of members minus 1).

■   RAIDsets and stripesets—the controller allocates the largest whole number of stripes that are less than or equal to the percentage you specify.

An unpartitioned storage unit has more capacity than a partition that uses the whole unit because each partition requires 5 blocks of administrative metadata. Thus, a single disk unit that contains one partition can store n minus 5 blocks of user or application data.

See "Configuring a Partition" page 4 , for information on manually
partitioning a storageset or single-disk unit.

Guidelines for Partitioning Storagesets and Disk Drives

Keep these points in mind as you plan your partitions:

■ Partitioned storagesets cannot function in multiple-bus failover
dual-redundant controller configurations.

■ If partitions were created prior to configuring a controller pair for
multiple-bus dual-redundant failover, then you must delete the
partitions before the configuring the controllers.

■ You can create up to eight partitions per storageset or disk drive.

■ All of the partitions on the same storageset or disk drive must be
addressed through the same target ID (host-addressable SCSI ID).
Thus, if you set a preferred controller for that ID, all the partitions
in that storageset will inherit that preferred controller. This ensures
a transparent failover of devices should one of the dual-redundant
controllers fail.

■ Partitions cannot be combined into storagesets. For example, you
cannot divide a disk drive into three partitions, then combine those
partitions into a RAIDset.

■ Partitioned storagesets cannot function in multiple bus failover
dual-redundant configurations. Because they are not supported, you
must delete your partitions before configuring the controllers for
multiple bus failover.

■ Once you partition a container, you cannot unpartition it without
reinitializing the container.

■ Just as with storagesets, you do not have to assign unit numbers to
partitions until you are ready to use them.

■ The CLONE utility cannot be used with partitioned mirrorsets or
partitioned stripesets.

Choosing Switches for Storagesets and Devices

Depending upon the kind of storageset or device being configured, you
can enable the following kinds of options or "switches":

■ RAIDset and mirrorset switches

■ INITIALIZE switches

■ ADD UNIT switches

■   ADD DEVICE switches

Enabling Switches

If you use the StorageWorks Command Console to configure the device or storageset, you can set switches from the command console screens during the configuration process. The Command Console automatically applies them to the storageset or device. See *Getting Started with Command Console*, for information about using the Command Console.

If you use CFMENU to configure the device or storageset, it prompts you for the switches during the configuration process and automatically applies them to the storageset or device.

If you use CLI commands to configure the storageset or device manually, the procedures in "Single Controller CLI Configuration Procedure" page 4 , indicate when and how to enable each switch.

Changing Switches

You can change the RAIDset, mirrorset, device, and unit switches at any time. See "Changing Switches for a Storageset or Device" page 4 , for information about changing switches for a storageset or device.

You cannot change the initialize switches without destroying the data on the storageset or device. These switches are integral to the formatting and can only be changed by re-initializing the storageset.

Note   Initializing a storageset is similar to formatting a disk drive; all of the data is destroyed during this procedure.

RAIDset Switches

You can enable the following kinds of switches to control how a RAIDset behaves to ensure data availability:

■   Replacement policy

■   Reconstruction policy

■   Membership

Replacement Policy

Specify a replacement policy to determine how the controller replaces a failed disk drive:

■ *POLICY=BEST_PERFORMANCE* (default) puts the failed disk drive in the failedset then tries to find a replacement (from the spareset) that is on a different device port than the remaining, operational disk drives. If more than one disk drive meets this criterion, this switch selects the drive that also provides the best fit.

■ *POLICY=BEST_FIT* puts the failed disk drive in the failedset then tries to find a replacement (from the spareset) that most closely matches the size of the remaining, operational disk drives. If more than one disk drive meets this criterion, this switch selects the one that also provides the best performance.

■ *NOPOLICY* puts the failed disk drive in the failedset and does not replace it. The storageset operates with less than the nominal number of members until you specify a replacement policy or manually replace the failed disk drive.

Reconstruction Policy

Specify the speed with which the controller reconstructs the data from the remaining operational disk drives and writes it to a replacement disk drive:

■ *RECONSTRUCT=NORMAL* (default) balances the overall performance of the subsystem against the need for reconstructing the replacement disk drive.

■ *RECONSTRUCT=FAST* gives more resources to reconstructing the replacement disk drive, which may reduce the overall performance of the subsystem during the reconstruction task.

Membership

Indicate to the controller that the RAIDset you are adding is either complete or reduced, which means it is missing one of its members:

■ *NOREDUCED* (default) indicates to the controller that all of the disk drives are present for a RAIDset.

■ *REDUCED* lets you add a RAIDset that is missing one of its members. For example, if you dropped or destroyed a disk drive while moving a RAIDset, you could still add it to the subsystem by using this switch.

Mirrorset Switches

You can enable the following switches to control how a mirrorset behaves to ensure data availability:

■   Replacement policy

■   Copy speed

■   Read source

Replacement Policy

Specify a replacement policy to determine how the controller replaces a failed disk drive:

■   *POLICY=BEST_PERFORMANCE* (default) puts the failed disk drive in the failedset then tries to find a replacement (from the spareset) that is on a different device port than the remaining, operational disk drives. If more than one disk drive meets this criterion, this switch selects the drive that also provides the best fit.

■   *POLICY=BEST_FIT* puts the failed disk drive in the failedset then tries to find a replacement (from the spareset) that most closely matches the size of the remaining, operational disk drives. If more than one disk drive meets this criterion, this switch selects the one that also provides the best performance.

■   *NOPOLIC*Y puts the failed disk drive in the failedset and does not replace it. The storageset operates with less than the nominal number of members until you specify a replacement policy or manually replace the failed disk drive.

Copy Speed

Specify a copy speed to determine the speed with which the controller copies the data from an operational disk drive to a replacement disk drive:

■   *COPY=NORMA*L (default) balances the overall performance of the subsystem against the need for reconstructing the replacement disk drive.

■   *COPY=FAS*T allocates more resources to reconstructing the replacement disk drive, which may reduce the overall performance of the subsystem during the reconstruction task.

Read Source

Specify the read source to determine how the controller reads data from the members of a mirrorset:

■ *READ_SOURCE=LEAST_BUSY* (default) forces the controller to read data from the "normal" or operational member that has the least-busy work queue.

■ *READ_SOURCE=ROUND_ROBIN* forces the controller to read data sequentially from all "normal" or operational members in a mirrorset. For example, in a four-member mirrorset (A, B, C, and D), the controller reads from A, then B, then C, then D, then A, then B, and so forth. No preference is given to any member.

■ *READ_SOURCE=DISKnnnnn* forces the controller to always read data from a particular "normal" or operational member. If the specified member fails, the controller reads from the least busy member.

Partition Switches

The geometry parameters of a partition can be specified. The geometry switches are:

■ *CAPACITY*—the number of logical blocks. The range is from 1 to the maximum container size.

■ *CYLINDERS*—the number of cylinders used in the partition. The range is from 1 to 16777215.

■ *HEADS*—the number of disk heads used in the partition. The range is from 1 to 255.

■ *SECTORS_PER_TRACK*—the number of sectors per track used in the partition. The range is from 1 to 255.

Device Switches

When you add a disk drive or other storage device to your subsystem, you can enable the following switches:

■ Transportability

■ Device Transfer rate

Transportability

The transportability switch indicates whether a disk drive is transportable or notransportable when you add it to your subsystem:

■ *NOTRANSPORTABLE* disk drives (default) are marked with StorageWorks-exclusive metadata. This metadata supports the error-detection and recovery methods that the controller uses to ensure data availability. Disk drives that contain this metadata cannot be used in non-StorageWorks subsystems.

Consider these points when using the *NOTRANSPORTABLE* switch:

– When you bring non-transportable devices from another subsystem to your controller subsystem, add the device to your configuration using the ADD command. Do not initialize the device or you will reset and destroy any forced error information contained on the device.

– When you add units, the controller software verifies that the disks or storagesets within the units contain metadata. To determine whether a disk or storageset contains metadata, try to create a unit from it. This causes the controller to check for metadata. If no metadata is present, the controller displays a message; initialize the disk or storageset before adding it.

■ *TRANSPORTABLE* disk drives can be used in non-StorageWorks subsystems. Transportable disk drives can be used as single-disk units (JBOD) in StorageWorks subsystems as well as disk drives in other systems. They **cannot** be combined into storagesets in a StorageWorks subsystem.

TRANSPORTABLE is especially useful for moving a disk drive from a workstation into your StorageWorks subsystem. When you add a disk drive as transportable, you can configure it as a single-disk unit and access the data that was previously saved on it.

Transportable devices have these characteristics:

– Can be interchanged with any SCSI interface that does not use the device metadata, for example, a PC.

– Cannot have write-back caching enabled.

– Cannot be members of a shadowset, storageset, or spareset.

– Do not support forced errors.

Consider these points when using the *TRANSPORTABLE* switch:

– Before you move devices from the subsystem to a foreign subsystem, delete the units and storagesets associated with the device and set the device as transportable. Initialize the device to remove any metadata.

– When you bring foreign devices into the subsystem with customer data follow this procedure:

1. Add the disk as a transportable device. Do not initialize it.

2. Copy the data the device contains to another nontransportable unit.

3. Initialize the device again after resetting it as nontransportable. Initializing it now places metadata on the device.

– Storagesets cannot be made transportable. Specify *NOTRANSPORTABLE* for all disks used in RAIDsets, stripesets, and mirrorsets.

– Do not keep a device set as transportable on a subsystem. The unit attached to the device loses forced error support which is mandatory for data integrity on the entire array.

Device Transfer Rate

Specify a transfer rate that the controller uses to communicate with the device. Use one of these switches to limit the transfer rate to accommodate long cables between the controller and a device, such as a tape library. Use one of the following values:

■ *TRANSFER_RATE_REQUESTED=20MHZ* (default)

■ *TRANSFER_RATE_REQUESTED=10MHZ*

■ *TRANSFER_RATE_REQUESTED=5MHZ*

■ *TRANSFER_RATE_REQUESTED=ASYNCHRONOUS*

Initialize Command Switches

You can enable the following kinds of switches to affect the format of a disk drive or storageset:

■ Chunk size (for stripesets and RAIDsets only)

■ Destroy/Nodestroy

■ Save configuration

■ Overwrite

After you initialize the storageset or disk drive, you cannot change these switches without reinitializing the storageset or disk drive.

Chunk Size

Specify a chunk size to control the stripesize used for RAIDsets and stripesets:

■ *CHUNKSIZE=DEFAULT* lets the controller set the chunk size based on the number of disk drives (d) in a stripeset or RAIDset. If $d \leq 9$ then chunk size = 256. If $d > 9$ then chunk size = 128.

■ *CHUNKSIZE=n* lets you specify a chunk size in blocks. The relationship between chunk size and request size determines whether striping increases the request rate or the data-transfer rate.

Tip   While a storageset may be initialized with a user-selected chunk size, it is recommended that only the default value of 128K be used. The default value is chosen to produce optimal performance for a wide variety of loads. The use of a chunk size less than 128 blocks (64K) is **strongly discouraged**. There are almost no customer loads for which small chunk sizes are of value and, in almost all cases, selecting a small chunk size will **severely** degrade the performance of the storageset **and** the controller as a whole. Use of a small chunk size on any storageset can result in **severe** degradation of overall system performance.

Increasing the Request Rate

A large chunk size (relative to the average request size) increases the request rate by allowing multiple disk drives to respond to multiple requests. If one disk drive contains all of the data for one request, then the other disk drives in the storageset are available to handle other requests. Thus, in principle, separate I/O requests can be handled in parallel, thereby increasing the request rate. This concept is shown in Figure 3–13.

Figure 3–13   Chunk Size Larger than the Request Size
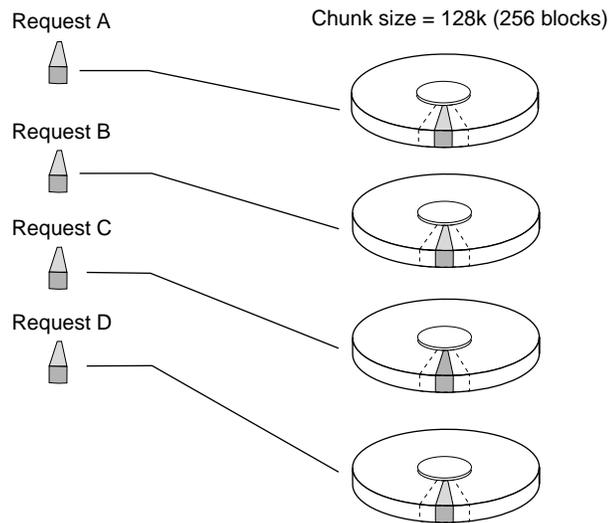
CXO-5135A-MC

Applications such as interactive transaction processing, office automation, and file services for general timesharing tend to require high I/O request rates.

Large chunk sizes also tend to increase the performance of random reads and writes. It is recommended that you use a chunk size of 10 to 20 times the average request size, rounded up to the nearest multiple of 64. In general, a chunk size of 256 works well for UNIX® systems; 128 works well for OpenVMS™ systems.

Increasing the Data Transfer Rate
A small chunk size relative to the average request size increases the data transfer rate by allowing multiple disk drives to participate in one I/O request. This concept is shown in Figure 3–14.

Figure 3–14   Chunk Size Smaller than the Request Size

Chunk size = 128k (256 blocks)
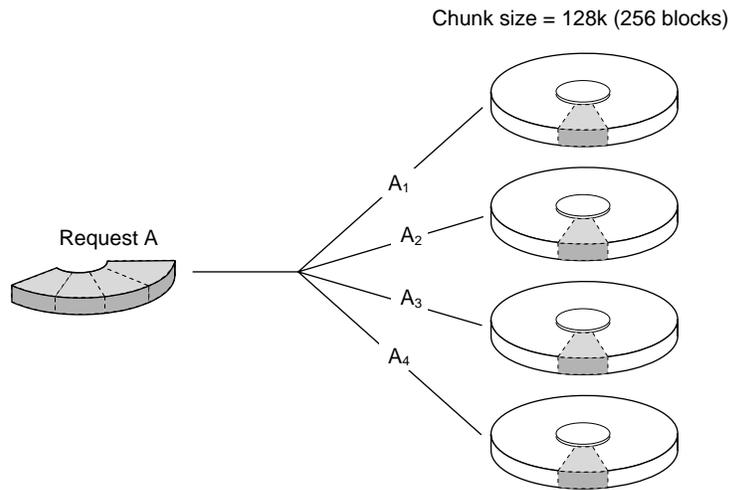
CXO-5172A-MC

Applications such as CAD, image processing, data collection and reduction, and sequential file processing tend to require high data-transfer rates.

Increasing Sequential Write Performance
For stripesets (or striped mirrorsets), use a large chunk size relative to the I/O size to increase the sequential write performance. A chunk size of 256 generally works well.

Chunk size does not significantly affect sequential read performance.

Maximum Chunk Size for RAIDsets
Do not exceed the chunk sizes shown in Table 3–2 for a RAIDset. (The maximum chunk size is derived by $2048/(d-1)$ where d is the number of disk drives in the RAIDset.)

Table 3–2   Maximum Chunk Sizes for a RAIDset

RAIDset Size	Max Chunk Size
3 members	1024 blocks
4 members	682 blocks
5 members	512 blocks
6 members	409 blocks
7 members	341 blocks
8 members	292 blocks
9 members	256 blocks
10 members	227 blocks
11 members	204 blocks
12 members	186 blocks
13 members	170 blocks
14 members	157 blocks

Save Configuration

Use the *SAVE_CONFIGURATION* switch to indicate whether to save the subsystem configuration on the storage unit when you initialize it:

Note   The *SAVE_CONFIGURATION* switch is recommended for single configurations only. Do not use for dual-redundant configurations. Use the CLI Command: SET FAILOVER COPY=.

■ *NOSAVE_CONFIGURATION* (default) means that the controller stores the subsystem configuration in its nonvolatile memory only. Although this is generally secure, the configuration could be jeopardized if the controller fails. For this reason, you should initialize at least one of your storage units with the *SAVE_CONFIGURATION* switch enabled.

■ *SAVE_CONFIGURATION* allows the controller to use 256K of each device in a storage unit to save the subsystem configuration. The controller saves the configuration every time you change it or add a patch to your controller. If the controller should fail, you can recover your latest configuration from the storage unit rather than rebuild it from scratch.

The save configuration option saves:

– All configuration information normally saved when you restart your controller except the controller serial number, product ID number, vendor ID number, and any manufacturing fault information.

– Patch information.

The save configuration option does not save:

– Software or hardware upgrades

– Inter-platform conversions

Consider the following when saving the configuration:

■ It is not necessary to use the *SAVE_CONFIGURATION* switch for dual-redundant configurations. Use the SET FAILOVER COPY=*controller* command to restore configuration information in a replacement controller. See *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for details.

■ Do not remove and replace disk devices between the time you save and restore your configuration. This is particularly important for devices that you migrate from another system. The controller could recover and use the wrong configuration information on your subsystem.

■ Save your subsystem configuration as soon as possible after removing and replacing any disk devices in your subsystem. This ensures that the devices always contain the latest, valid information for your system.

■ When you incorporate a spare into a storageset that you initialized with the INITIALIZE *SAVE_CONFIGURATION* command, the controller reserves space on the spare for configuration information. The controller updates this information when the configuration changes.

■ You cannot use a storageset that contains user data to save your subsystem configuration unless you backup and restore the user data.

■ If you previously configured storagesets with the *SAVE_CONFIGURATION* switch, you do not need to initialize them again after you reconfigure your devices with a new controller.

■ When you replace a controller, make sure the replacement controller does not contain any configuration data. If the controller is not new, initialize it with the SET *this_controller INITIAL_CONFIGURATION* command. If you do not take this precaution, you can lose configuration data if non-volatile memory changes.

Destroy/Nodestroy

Specify whether to destroy or retain the user data and metadata when you initialize a disk drive that has been previously used in a mirrorset or as a single-disk unit.

Note   The *DESTROY* and *NODESTROY* switches are only valid for striped mirrorsets and mirrorsets.

■ *DESTROY* (default) overwrites the user data and forced-error metadata on a disk drive when it is initialized.

■ *NODESTROY* preserves the user data and forced-error metadata when a disk drive is initialized. Use *NODESTROY* to create a single-disk unit from any disk drive that has been used as a member of a mirrorset. See the REDUCED command in the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for information on removing disk drives from a mirrorset.

   NODESTROY is ignored for members of a RAIDset, all of which are destroyed when the RAIDset is initialized.

ADD UNIT Switches

You can enable the ADD UNIT switches listed in Table 3–3 for the listed storagesets and devices. Explanations of these switches follow the table. See the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for a complete list of ADD UNIT switches.

Table 3–3   ADD UNIT Switches for New Containers

Switch	RAIDset	Stripeset	Mirrorset	Non-Transportable Disk	Transportable Disk	CD-ROM	Partition	Passthrough
ACCESS_ID=ALL *ACCESS_ID=unit identification*	✓	✓	✓	✓	✓	✓	✓	
PARTITION=partition-number	✓	✓	✓	✓	✓	✓	✓	
MAXIMUM_CACHED_TRANSFER	✓	✓	✓	✓	✓	✓	✓	
PREFERRED_PATH *NOPREFERRED_PATH*	✓	✓	✓	✓	✓	✓	✓	✓
READ_CACHE *NOREAD_CACHE*	✓	✓	✓	✓	✓	✓	✓	
RUN *NORUN*	✓	✓	✓	✓	✓	✓		
WRITE_PROTECT *NOWRITE_PROTECT*	✓	✓	✓	✓	✓		✓	
WRITEBACK_CACHE *NOWRITEBACK_CACHE*	✓	✓	✓	✓			✓	

Access Protection
Restrict or remove host access restrictions for a storage unit:

■   *ACCESS_ID*=ALL (default) allows all hosts to access all units.

■   *ACCESS_ID=nn* allows you to specify an access ID for a unit that
corresponds to a SCSI host ID. Specifying an ID indicates that the
unit can be accessed only by the host with the specified target ID
number.

In a multiple host environment specifying access IDs prevents multiple
hosts from accessing the same unit. Use this feature on a unit-by-unit
basis to ensure units are accessed only by the host for which they are
configured.

The access ID for a passthrough unit is always ALL; you cannot
change the access ID for a passthrough unit.

You can define a unit access ID with the ADD command when
presenting the unit to the host, or you can specify an access ID for a
unit already presented to the host, or change a unit access ID, by using
the SET command. See the *HSZ70 Array Controller HSOF Version 7.3
CLI Reference Manual* for more information about using the ADD and
SET commands to define an access ID.

See the *StorageWorks Array Controller HSZ70 Array Controller
Operating Software HSOF Version 7.3 Release Notes* for configuration
rules and requirements that you should consider when setting up and
working in a multiple host environment.

Partition
Specify the partition number that identifies the partition associated with
the host-addressable unit number you are adding. Partitioned units must
have the same SCSI target ID number and must be part of the same
container.

PARTITION=partition-number allows you to identify a partition that is
associated with the unit you're adding.

Maximum Cache Transfer
Specify the amount of data (in blocks) that the controller may cache to
satisfy a read request:

■   *MAXIMUM_CACHED_TRANSFER=n* lets you indicate the
number of data blocks that the controller will cache to satisfy a read
request. Any I/O transfers in excess of the specified size will not be
cached. You can specify a value from 1 to 1024.

■ *MAXIMUM_CACHED_TRANSFER*=32 (default) is the default number of data blocks that the controller will cache to satisfy a read request.

The *MAXIMUM_CACHED_TRANSFER* switch affects both read and write-back cache when set on a controller that has read and write-back caching.

Preferred Path

The *PREFERRED PATH* switch applies only to containers in a multiple-bus failover configuration. See the discussion about assigning unit numbers in the section "Assigning Unit Numbers and Unit Qualifiers," page 4-25 to find out how you can use unit numbers to establish preferred paths for storage units in a dual-redundant configuration.

Specify which controller accesses the container. If one controller fails, the operational controller handles the I/O activity to all of the storage units regardless of their preferred paths:

■ *NOPREFERRED_PATH* (default) allows either controller to access the storage unit.

■ *PREFERRED_PATH=this_controller* indicates that the controller to which you are connected handles all I/O activity to the storage unit. By establishing preferred paths, you can distribute the I/O load evenly between the two controllers by dividing the storage units into two equal groups—and assigning each group to its own controller.

■ *PREFERRED_PATH=other_controller* indicates that the other controller—the one to which you are not connected—handles all I/O activity to the storage unit.

Read Cache

Enable or disable the caching of read data to the container:

■ *READ_CACHE* (default) enables the caching of read data.

■ *NOREAD_CACHE* disables the caching of read data.

Availability

Specify whether or not to make the container available to the host. This switch is not valid for partitioned units. Do not specify this switch on the SET or ADD commands for a partitioned unit.

■ *RUN* (default) specifies that as soon as you provide a host-addressable unit number the storage unit will be made available to the host.

■ *NORUN* specifies that the container will not be made available to the host until you specify the RUN switch.

Write Protection

Enable or disable write protection for the container:

■ *NOWRITE_PROTECT* (default) enables the controller to write new data to the storage unit.

■ *WRITE_PROTECT* prevents the controller from writing any new data to the storage unit. (The controller can write to a protected unit if it needs to reconstruct data.)

Write-back Cache

Enable or disable the controller write-back caching for a container:

■ *WRITEBACK_CACHE* enables write-back caching.

■ *NOWRITEBACK_CACHE* (default) disables write-back caching.

Note   If you disable write-back caching for a storage unit that previously used it, it may take up to five minutes to flush the unwritten data from the cache to the devices in the storage unit.

Storage Maps

Configuring your subsystem will be easier if you know how the storagesets, partitions, and JBODs correspond to the disk drives in your subsystem. You can more easily see this relationship by creating a hardcopy representation (a storage map). Figure is a representative blank storage map showing a simplified physical representation of the enclosure (each cell in the map represents a disk drive in the enclosure). The location of the drive determines the PTL location (Figure 3–15).

Creating a Storage Map

If you want to make a storage map, fill out the blank storage map (shown in Figure 3–15) as you add storagesets, partitions, and JBOD disks to your configuration and assign them unit numbers. Label each disk drive in the map with the higher levels it is associated with, up to the unit level.

Appendix B contains blank templates you may use in the creation of your subsystem storage map.

Figure 3–15   Example Blank Storage Map

Power Supply							Power Supply
	Disk10300	Disk20300	Disk30300	Disk40300	Disk50300	Disk60300	
Power Supply							Power Supply
	Disk10200	Disk20200	Disk30200	Disk40200	Disk50200	Disk60200	
Power Supply							Power Supply
	Disk10100	Disk20100	Disk30100	Disk40100	Disk50100	Disk60100	
Power Supply							Power Supply
	Disk10000	Disk20000	Disk30000	Disk40000	Disk50000	Disk60000	

Example Storage Map

Figure 3–16 is an example of a completed storage map in a single-enclosure subsystem with the following configured storage:

- Unit D100 is a 6-member RAID 3/5 storageset named R1. R1 consists of Disk10000, Disk2000, Disk 30000, Disk 40000, Disk 50000, and Disk 60000.

- Unit D101 is a 6-member RAID 3/5 storageset named R2. R2 consists of Disk 10100, DIsk 20100, Disk 30100, Disk 40100, DIsk 50100, and Disk 60100.

- Unit D102 is a 2-member striped mirrorset named S1. S1 consists of M1 and M2:

  – M1 is a 2-member striped mirrorset consisting of Disk 10200 and Disk 20200.

  – M2 is a 2-member mirrorset consisting of Disk30200 and Disk40200.

- Unit D103 is a 2-member mirrorset named M3. M3 consists of Disk50200 and Disk60200.

- Unit D104 is a 4-member stripeset named S2. S2 consists of Disk10300, Disk20300, Disk 20300, and Disk 40300.

- Unit D105 is a single (JBOD) disk named Disk50300.

- Disk 60300 is an autospare.

Figure 3–16   Completed Example Storage Map

Power Supply	D104 S2 Disk10300	D104 S2 Disk20300	D104 S2 Disk30300	D104 S2 Disk40300	D105 Disk50300	Autospare Disk60300	Power Supply
Power Supply	D102 S1 M1 Disk10200	D102 S1 M1 Disk20200	D102 S1 M2 Disk30200	D102 S1 M2 Disk40200	D103 M3 Disk50200	D103 M3 Disk60200	Power Supply
Power Supply	D101 R2 Disk10100	D101 R2 Disk20100	D101 R2 Disk30100	D101 R2 Disk40100	D101 R2 Disk50100	D101 R2 Disk60100	Power Supply
Power Supply	D100 R1 Disk10000	D100 R1 Disk20000	D100 R1 Disk30000	D100 R1 Disk40000	D100 R1 Disk50000	D100 R1 Disk60000	Power Supply

Using the LOCATE Command to Find Devices

If you want to complete a storage map at a later time but don't remember where everything is, use the CLI command LOCATE. The LOCATE command flashes the amber (fault) LED on the drives associated with the specific storageset or unit. To turn off the flashing LEDs, enter the CLI command LOCATE *cancel*.

The following is an example of the commands needed to locate all the drives that make up D104:

LOCATE D104

After you have noted the position of all the device contained within D104, enter the following to turn off the flashing LEDs:

LOCATE CANCEL

The following is an example of the commands needed to locate all the drives that make up RAIDset R1:

LOCATE R1

After you have noted the position of the devices contained within R1, enter the following to turn off the flashing LEDs:

LOCATE CANCEL

Moving Containers

You can move a container from one subsystem to another without destroying its data. You can also move a container to a new location within the same subsystem. The following sections describe the rules to be observed when moving containers:

■ HSZ70 Array Controllers and Asynchronous Drive Hot Swap

■ Container Moving Procedures

HSZ70 Array Controllers and Asynchronous Drive Hot Swap

Asynchronous Drive Hot Swap (ADHS) is defined as the removal or insertion of a drive without quiescing the bus. Note that disk replacement within storagesets is performed by sparing, either by using the AUTOSPARE feature, or by using manual intervention with CLI commands to delete and replace the device.

ADHS is supported on the HSZ70 with the observance of the following restrictions:

■ Applies to disk drives only (wait 90 seconds after return of power on before enabling the bus, issuing CLI commands to the controller, and all activity to it).

■ Disks may not be imported into slots configured as disks which are members of higher-level containers such as RAIDsets, mirrorsets, sparesets, and so on. AUTOSPARING is used for these types of configurations.

ADHS is not supported under the following operating conditions:

■ During failover.

■ During failback.

■ During controller initialization/reboot (until the CLI prompt appears).

■ During the running of a local program (DILX, CLCP, and so on).

■   To perform a physical move of a device from one location to another (new port or target).

■   To perform more than one drive removal/insertion at a time (50 seconds of time is required for the controller to complete the process of recognizing/processing the drive removal/insertion.

Note   When power cycling entire shelves during servicing, ensure all controller-based Mirror/RAID drives have not ben moved to the failedset or are faulted.

Container Moving Procedures

Use the following procedure to move a container while maintaining the data it contains (reference Figure 3–17):

Figure 3–17   Moving a Container from one Subsystem to Another
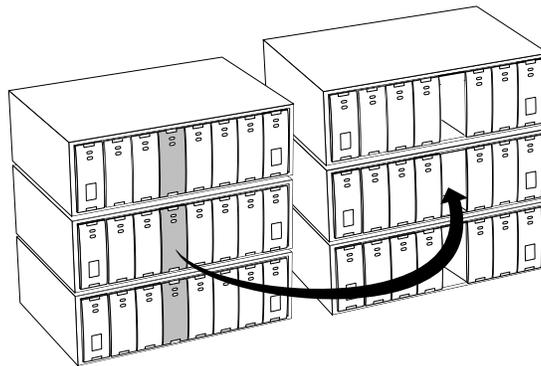
CXO5595A

⚠   **Caution**   Never initialize any container or this procedure will not protect data.

---

1.   Show the details for the container you want to move. Use the following syntax:

SHOW *storageset-name*

2. Label each member with its name and PTL location.

   If you do not have a storageset map for your subsystem, you can enter the LOCATE command for each member to find its PTL location. Use the following syntax:

   LOCATE *disk-name*

   To cancel the locate command, enter the following:

   LOCATE CANCEL

3. Delete the unit-number shown in the "Used by" column of the SHOW storageset-name command. Use the following syntax:

   DELETE *unit-number*

4. Delete the storageset shown in the "Name" column of the SHOW storageset-name command. Use the following syntax:

   DELETE *storageset-name*

5. Delete each disk drive—one at a time—that the storageset contained. Use the following syntax:

   DELETE *disk-name*
   DELETE *disk-name*
   DELETE *disk-name*

6. Remove the disk drives and move them to their new PTL locations.

7. Add again each disk drive to the controller list of valid devices. Use the following syntax:

   ADD DISK *disk-name PTL-location*
   ADD DISK *disk-name PTL-location*
   ADD DISK *disk-name PTL-location*

8. Recreate the storageset by adding its name to the controller list of valid storagesets and specifying the disk drives it contains. (Although you have to recreate the storageset from its original disks, you do not have to add them in their original order.) Use the following syntax:

   ADD *storageset-name disk-name disk-name*

9. Represent the storageset to the host by giving it a unit number the host can recognize. You can use the original unit number or create a new one. Use the following syntax:

   ADD UNIT *unit-number storageset-name*

Example 3–1
The following example moves unit D100 to another cabinet. D100 is
the RAIDset RAID99 that comprises members 20000, 30000, and
40000.

SHOW RAID99

NAME	STORAGESET	USES	USED BY
RAID99	RAIDSET	DISK10000	D100
		DISK20000	
		DISK30000	

DELETE D100
DELETE RAID99
DELETE DISK10000
DELETE DISK20000
DELETE DISK30000
(...move the disk drives to their new location...)
ADD DISK DISK20000 2 0 0
ADD DISK DISK30000 3 0 0
ADD DISK DISK40000 4 0 0
ADD RAIDSET RAID99 DISK20000 DISK30000 DISK40000
ADD UNIT D100 RAID99

Example 3–2

The following example moves the reduced RAIDset, R3, to another cabinet. (R3 used to contain DISK20000, which failed before the RAIDset was moved. R3 contained DISK10000, DISK30000, and DISK40000 at the beginning of this example.)

DELETE D100

DELETE R3

DELETE DISK10000

DELETE DISK30000

DELETE DISK40000

(...move disk drives to their new location...)

ADD DISK DISK10000 1 0 0

ADD DISK DISK30000 3 0 0

ADD DISK DISK40000 4 0 0

ADD RAIDSET R3 DISK10000 DISK30000 DISK40000 REDUCED

ADD UNIT D100 R3

The Next Step...

The preferred method for configuring your subsystem is to use the StorageWorks Command Console (SWCC). The manual *Getting Started with Command Console,* provides a description of the application and how to use it to configure storage subsystems. SWCC is a Graphical User Interface (GUI) used for the set-up and management of RAID storage subsystems.

Turn to Chapter 4, "Subsystem Configuration Procedures,", if you want to configure your storage units manually by issuing CLI commands from a local or remote connection. Configuring your storage units manually gives you control when it comes to naming the storage units. You should have completed a storageset profile for each storage unit/ device that you want configured in your subsystem before you begin these procedures.

CHAPTER 4

Subsystem Configuration Procedures

If you plan on configuring your subsystem using CLI commands instead of using the SWCC, then use the appropriate sections contained in this chapter:

- "Establishing a Local Connection to the Controller," page 4-2
- "Configuration Procedure Flowchart," page 4-5
- "Configuring a Single Controller," page 4-7
- "Configuring for Transparent Failover Mode," page 4-9
- "Configuring for Multiple-Bus Failover Mode," page 4-14
- "Configuring Devices," page 4-18
- "Configuring a Stripeset," page 4-19
- "Configuring a Mirrorset," page 4-20
- "Configuring a RAIDset," page 4-21
- "Configuring a Striped Mirrorset," page 4-22
- "Configuring a Single Disk Unit," page 4-22
- "Configuring a Partition," page 4-23
- "Assigning Unit Numbers and Unit Qualifiers," page 4-25
- "Preferring Units in Multiple-Bus Failover Mode," page 4-26
- "Configuration Options," page 4-26

Before starting, ensure you have completed the storage subsystem planning which was described in Chapters 2 and 3.

Establishing a Local Connection to the Controller

In the process of configuring your controller, you must be able to communicate with your controller either locally or remotely:

■   Use a local connection to configure the controller for the first time.

■   Use a remote connection to your host system for all subsequent configuration tasks.

See the *Getting Started* guide that came with your platform kit for details.

The maintenance port on the front of the HSZ70 (see Figure 4–1) provides a convenient place to connect a terminal to the controller so that configuration and troubleshooting tasks may be accomplished. This port accepts a standard RS-22 jack from any EIA-423 compatible terminal (or a PC with a terminal-emulation program). The maintenance port supports serial communications with default values of 9600 baud using 8 data bits, 1 stop bit, and no parity.

⚠   **Caution**   The local-connection port described in this book generates, uses, and can radiate radio-frequency energy through cables that are connected to it. This energy may interfere with radio and television reception. Do not leave any cables connected to it when you are not communicating with the controller.

Local Connection Procedures

Use the following procedures to establish a local connection for setting the controller's initial configuration. Turn off the terminal and connect the required cable(s) to the controller maintenance port as shown in Figure 4–1.

Note   The cable that is currently being shipped (17-04074-04; shown in Figure 4–1), does not support a direct connection to a terminal, some desktop or laptop PCs. If a connection is needed that is not supported by the shipped cable, then the parts shown in Figure 4–2 will need to be ordered.

Figure 4–1   Terminal to Maintenance Port Connection

CXO6484B
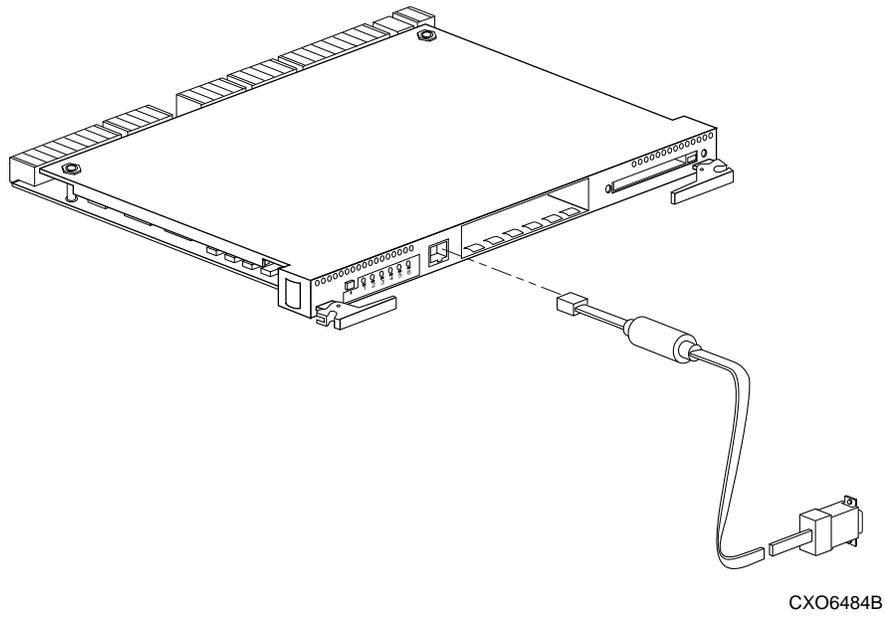
Figure 4–2   Optional Maintenance Port Connection Cabling

CXO6485A
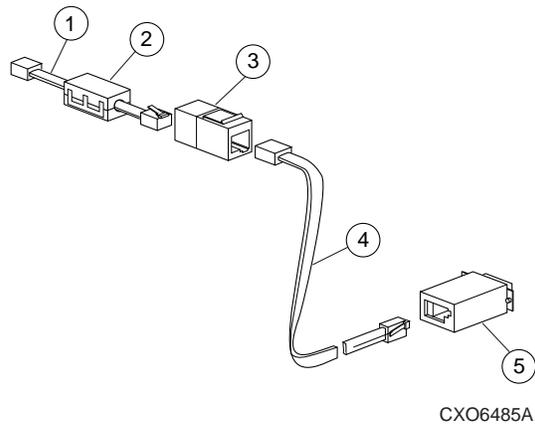
Table 4–1   Key to Figure 4–2

Item #	Description	Used On	Part No.
①	BC16E-xx Cable Assy	All	17-04074-01
②	Ferrite bead	All	16-25105-14
③	RJ-11 Adapter	All	12-43346-01
④	RJ-11 Extension Cable	All	17-03511-01
⑤	PC Serial Port Adapter: ■ 9 pin D-sub to 25 pin SKT D-sub ■ 9 pin D-sub to 25 pin D-sub ■ 9 pin D-sub to 25 pin D-sub, mod	PC Sun HP800	12-45238-01 12-45238-02 12-45238-03

1. Use Table 4–2 to choose the cabling method based upon whether the local connection is a PC or a maintenance terminal:

Table 4–2   PC/Maintenance Terminal Selection

If	Then	And
Maintenance Terminal	■ Plug one end of a DECconnect Office Cable (BC16E–XX) into the controller local connection port ■ Place the Ferrite Bead on the BC16E ■ Plug the other end into the RJ–11 adapter (12–43346–01) ■ Use the RJ–11 extension (17–03511–04) to connect the adapter to the maintenance terminal	Go to step 2
PC	■ Plug one end of the Maintenance Port Cable (17-04074-04) into the terminal ■ Plug the 9-pin connector end into the back of the PC	Go to step 2

2.   Turn on the terminal (or PC)

3.   Configure the terminal (or PC) for 9600 baud, 8 data bits, 1 stop bit, and no parity.

4.   Press the Enter or Return key. A copyright notice and the CLI prompt appear, indicating that a local connection has been made with the controller.

5.   Optional Step: Configure the controller to a baud rate of 19200:

   SET THIS_CONTROLLER TERMINAL SPEED=19200
   SET OTHER_CONTROLLER TERMINAL SPEED=19200

6.   Configure the terminal/PC for a baud rate of 19200

7.   Optional Step (to be completed if Step 5 is also completed): Configure the terminal (or PC) to a baud rate of 19200.

Note   Terminal Port connections at baud rates of 1800, 9600, and 19200 are supported by HSOF V7.3. Terminal port connections of 300, 1200, and 2400 are not supported.

Configuration Procedure Flowchart

Before you configure a controller and its storage array, you must have considered what kinds of storagesets you will need. Chapter 2 contains the many rules and other considerations regarding subsystem configuration, while the different types of storagesets were described in Chapter 3.
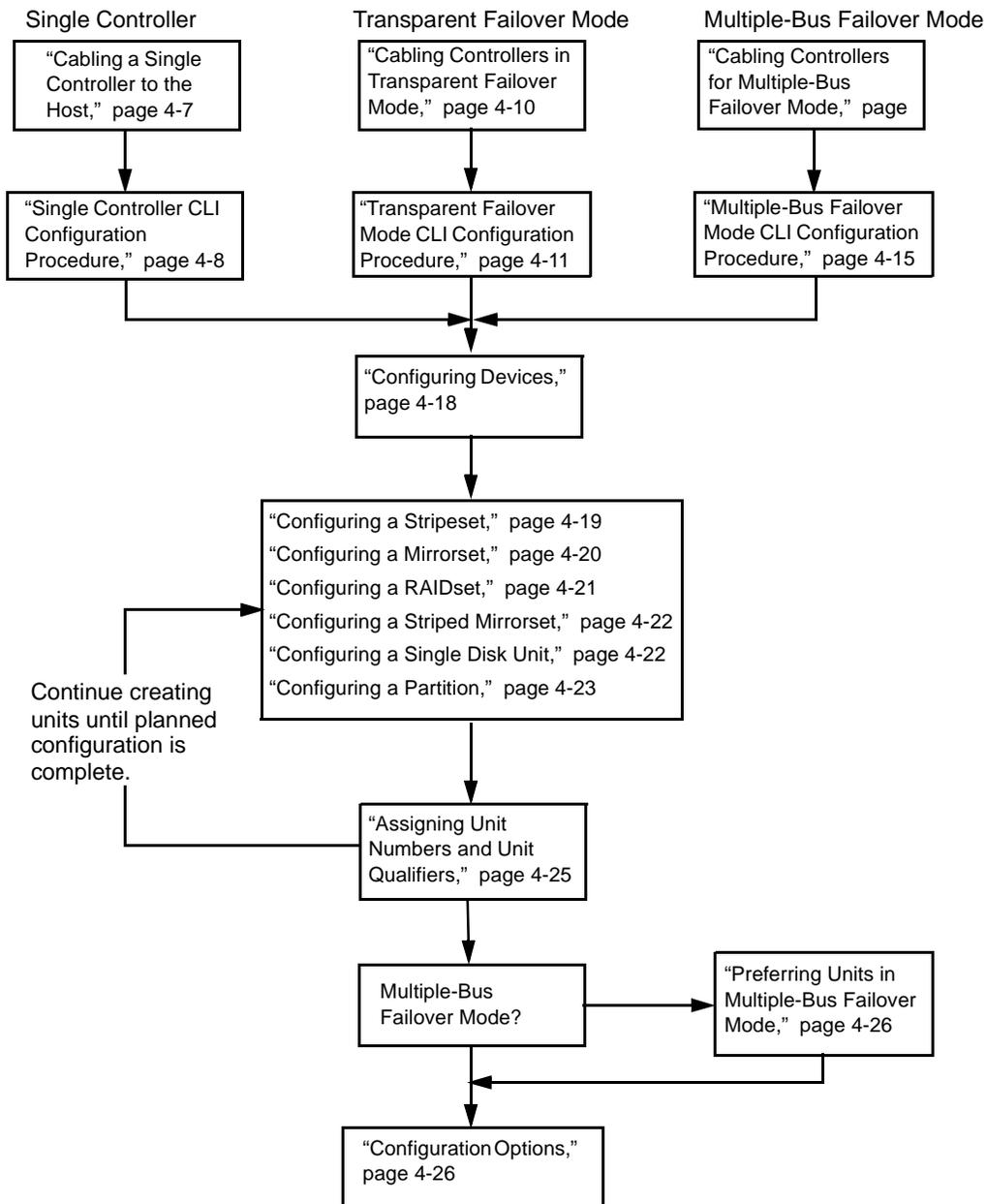
If all the prior storageset planning is completed, then continue on by using "The Configuration Flow Process," page 4-6.

Note   The following pages assume that a full controller/storageset configuration needs to take place (including the cabling of a new controller to the subsystem).

Figure 4–3   The Configuration Flow Process

Single Controller

> "Cabling a Single Controller to the Host," page 4-7

Transparent Failover Mode

> "Cabling Controllers in Transparent Failover Mode," page 4-10

Multiple-Bus Failover Mode

> "Cabling Controllers for Multiple-Bus Failover Mode," page

> "Single Controller CLI Configuration Procedure," page 4-8

> "Transparent Failover Mode CLI Configuration Procedure," page 4-11

> "Multiple-Bus Failover Mode CLI Configuration Procedure," page 4-15

> "Configuring Devices," page 4-18

> "Configuring a Stripeset," page 4-19
> "Configuring a Mirrorset," page 4-20
> "Configuring a RAIDset," page 4-21
> "Configuring a Striped Mirrorset," page 4-22
> "Configuring a Single Disk Unit," page 4-22
> "Configuring a Partition," page 4-23

Continue creating units until planned configuration is complete.

> "Assigning Unit Numbers and Unit Qualifiers," page 4-25

Multiple-Bus Failover Mode?

> "Preferring Units in Multiple-Bus Failover Mode," page 4-26

> "Configuration Options," page 4-26

Configuring a Single Controller

Configuring a single controller consists of cabling the controller to the host(s) and entering the configuration parameters.

Cabling a Single Controller to the Host

To connect a single, nonredundant controller to the host:

1. Stop all I/O from the host to its devices on the bus to which you are connecting the controller.

2. Refer to Figure 4–4 for the remainder of these steps: Remove the trilink connector (① 12–39921–01) from the controller. This connector is a 68-pin Y-adapter that maintains bus continuity even when it is disconnected from the controller.

3. Connect the bus cable ② from the host to one of the connectors on the front of the trilink connector.

4. If you're connecting a host to a controller in a BA370 enclosure that will reside in an SW600 cabinet, snap the ferrite bead ③ on the bus cable within one inch of the controller.

   If you are connecting a host to a controller in any other enclosure or cabinet, skip to step 5.

Figure 4–4   Connecting a Single Controller to Its Host

CXO6486B
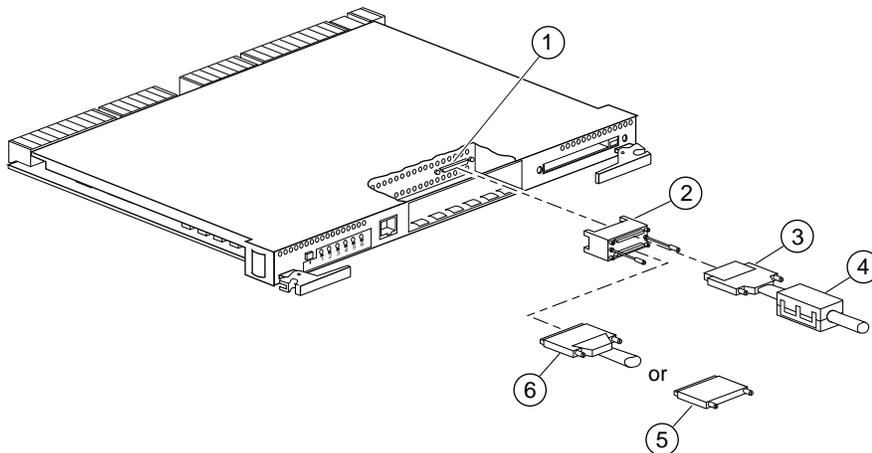
5.   Perform one of the following:

■   If the controller is at the end of the host bus, connect a terminator ④ to the other connector on the front of the trilink connector.

■   If the controller is not at the end of the host bus, connect a cable ⑤ that continues to the next device on the bus, and install the terminator at the end of the bus.

6.   Reconnect the trilink connector to the host port on the controller. Do not disconnect the host cables from the trilink connector.

7.   Route and tie the cables as desired.

Restart the I/O from the host. Some operating systems may require you to restart the host to see the devices attached to the new controller

Single Controller CLI Configuration Procedure

1.   Apply power to the subsystem.

The powerup sequence takes approximately 45 seconds. At the end of the powerup sequence, the audible alarm on the EMU will sound and the EMU error LED will be solidly lit. Turn off the alarm by pressing and releasing the EMU reset button once.

2.   Ensure the Power Verification and Addressing (PVA) module ID switch(es) in the array subsystem are set to the following:

■   Controller enclosure—set the PVA switch in that cabinet to 0.

■   The next enclosure in an extended controller subsystem—set the PVA switch in that cabinet to 2.

3.   The third cabinet in an extended controller subsystem—set the PVA switch in that cabinet to 3.

4.   Assign controller targets to the host ports according to the planned configuration. For example, to assign targets 1 and 2 use the SET *controller* command with the *ID=* switch:

SET THIS_CONTROLLER ID=1,2

This command requires the controller to restart before taking affect.

5.   Restart the controller by using the RESTART *controller* command:

RESTART THIS_CONTROLLER

It takes about one minute for the CLI prompt to come back after a RESTART command.

6. Set the time on the controller by using the SET *controller* command with the *TIME=* switch:

    SET THIS_CONTROLLER TIME= DD-MMM-YYYY:HH:MM:SS

7. Use the FRUTIL utility to set up the battery discharge timer:

    RUN FRUTIL

    Note   The RESTART THIS_CONTROLLER command begins a series of memory tests which take a few minutes to run. If the diagnostics are still running, FRUTIL will not start and a message informs you that memory tests are running. FRUTIL starts after the tests complete.

    Enter "Y" to the following FRUTIL display:
    ```
    Do you intend to replace this controller's cache
    battery? Y/N [N]
    ```

    FRUTIL prints out a procedure, but won't give you a prompt. Ignore the procedure and press Enter.

8. Set up any additional optional controller settings (for example, changing the CLI prompt). See the SET *this_controller* (SET *other_controller*) command in the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for the format of the options.

9. Enter a SHOW *this_controller* command to verify that all changes have taken place:

    SHOW THIS_CONTROLLER

Configuring for Transparent Failover Mode

Configuration procedures for a dual-redundant controller pair operating in transparent failover mode consists of cabling the controllers to the host(s) and entering the CLI configuration commands.

Cabling Controllers in Transparent Failover Mode

To connect a pair of dual-redundant controllers to the host:

1. Stop all I/O from the host to its devices on the bus to which you are connecting the controllers.

2. Refer to Figure 4–5 for the balance of these steps: Remove the trilink connectors (① 12–39921–01) from both controllers. These connectors are 68-pin Y-adapters that maintain bus continuity even when they are disconnected from their controller.

3. Connect the bus cable ② from the host to one of the connectors on the front of one of the trilink connectors.

4. If you are connecting a host to a controller in a BA370 enclosure that will reside in an SW600 cabinet, snap the ferrite bead ③ on the bus cable within one inch of the controller.

   If you are connecting a host to a controller in any other enclosure or cabinet, skip to step 5.

Figure 4–5   Connecting Dual-Redundant Controllers to the Host

CXO5612B
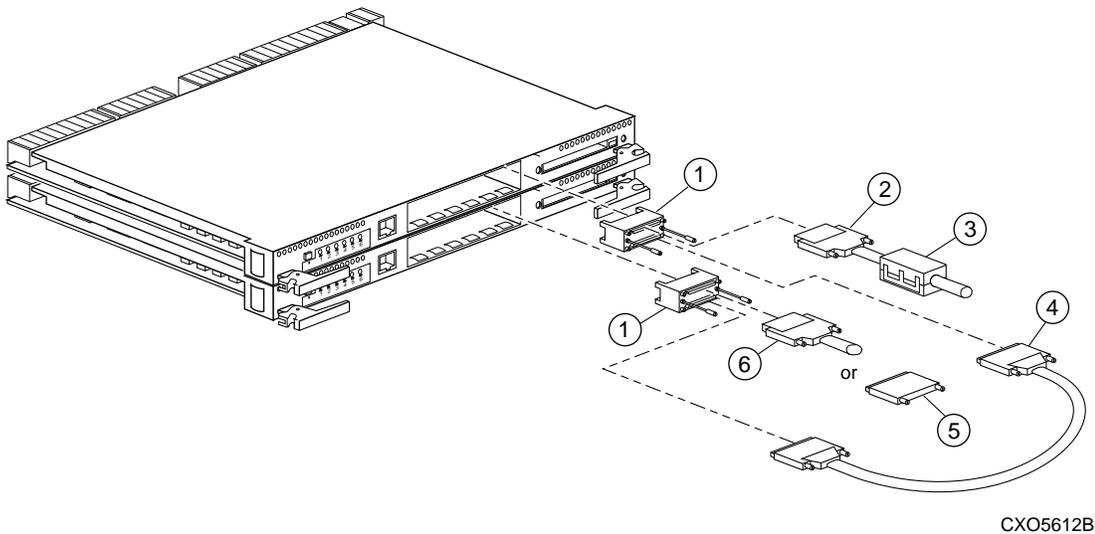
5. Connect the two trilink connectors with a jumper cable ④.

6. Perform one of the following:

■   If the controllers are at the end of the bus, connect a terminator ⑤ to the open connector on the front of the trilink connector.

■   If the controllers are not at the end of the bus, connect a cable ⑥ that continues to the next device on the bus and install the terminator at the end of the bus.

7.   Reconnect the trilink connectors to the host ports on the controllers. Do not disconnect the host cables from the trilink connector.

8.   Route and tie the cables as desired.

Restart the I/O from the host. Some operating systems may require you to restart the host to see the devices attached to the new controller.

Transparent Failover Mode CLI Configuration Procedure

1.   Apply power to the subsystem.

The powerup sequence takes approximately 45 seconds. At the end of the powerup sequence, the audible alarm on the EMU will sound and the EMU error LED will be solidly lit. Turn off the alarm by pressing and releasing the EMU reset button once.

Note   The audible alarm will cease, but the LED remains on until the controllers are bound into failover mode. The CLI will display a copyright notice and a report for the "other" controller.

---

2.   Ensure the Power Verification and Addressing (PVA) module ID switch(es) in the array subsystem are set to the following:

■   Controller enclosure—set the PVA switch in that cabinet to 0.

■   The next enclosure in an extended controller subsystem—set the PVA switch in that cabinet to 2.

■   The third cabinet in an extended controller subsystem—set the PVA switch in that cabinet to 3.

3.   Enter the following command to stop the CLI from reporting a misconfiguration error resulting from having no failover mode specified:

CLEAR CLI

4.   Ensure that neither of the controllers is already placed into a transparent failover mode by using the SET NOFAILOVER command:

SET NOFAILOVER

5. Place the controller pair into transparent failover mode with the SET FAILOVER COPY=*controller* command:

SET FAILOVER COPY=THIS_CONTROLLER

The copy qualifier normally specifies where the good copy of the array configuration is. At this point in the configuration process there is no array configuration, but the qualifier must be specified as it is part of the command syntax.

Note   When the command is entered, both controllers restart. The restart may set off the EMU audible alarm. To silence the alarm, press the EMU alarm button once and release. The audible alarm will cease, but the LED remains on until the controllers bind into failover mode. This process takes about 15 seconds.

The CLI will print out a report from the "other" controller indicating the "other" controller restarted. The CLI will continue reporting this condition until cleared with the following command:

CLEAR CLI

6. Set up mirrored cache (if desired) for the controller pair using the SET *controller* with the *MIRRORED_CACHE* switch:

SET THIS_CONTROLLER MIRRORED_CACHE
SET OTHER_CONTROLLER MIRRORED_CACHE

This command causes a restart, so the EMU audible alarm may sound.

7. Assign controller targets to the host port according to the planned configuration. For example, to assign targets 1 and 2 use the SET *controller* command with the *ID=* switch:

SET THIS_CONTROLLER ID=1,2

This command requires both controllers to restart before taking effect.

8. Restart the "other" controller first, then "this" controller with the RESTART *controller* command:

RESTART OTHER_CONTROLLER
RESTART THIS_CONTROLLER

> **Note**   When the "other" controller restarts, the EMU audible alarm sounds, but ceases when "this" controller restarts.

It takes about one minute for the CLI prompt to come back after the issuance of a RESTART command.

9. If desired, prefer targets to "this" controller. As an example, to prefer target 2 to "this" controller, use the SET *controller* command with the *PREFERRED_ID=* switch:

    SET THIS_CONTROLLER PREFERRED_ID=2

10. Set the time on the controller with the SET *controller* command with the *TIME=* switch:

    SET THIS_CONTROLLER TIME= DD-MMM-YYYY:HH:MM:SS

11. Use the FRUTIL utility to set up the battery discharge timer:

    RUN FRUTIL

> **Note**   The RESTART *this_controller* command begins a series of memory tests which take a few minutes to run. If the diagnostics are still running, FRUTIL will not start and a message informs you that memory tests are running. FRUTIL starts after the tests complete.

Enter "Y" to the following FRUTIL display:
```
Do you intend to replace this controller's cache
battery? Y/N [N]
```

FRUTIL prints out a procedure, but won't give you a prompt. Ignore the procedure and press Enter.

12. Move the maintenance cable to the "other" controller, and repeat steps 8 and 9.

13. Set up any additional optional controller settings (for example, changing the CLI prompt). See the SET *this_controller* (SET *other_controller*) command in the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for the format of the options.

14. Enter a SHOW *this_controller* and a SHOW *other_controller* command to verify that all changes have taken place.

Configuring for Multiple-Bus Failover Mode

Conceptually, the multiple-bus failover (host-assisted) configuration is shown in Figure 2–5. Physically, The configuration procedure for a dual-redundant controller pair operating in multiple-bus failover mode consists of cabling the controllers to the host(s) and entering the CLI configuration commands.Cabling and configuring the ntrollers is described in the paragraphs that follow.

Cabling Controllers for Multiple-Bus Failover Mode

To connect a pair of multiple bus failover dual-redundant controllers to the host:

1. Stop all I/O from the host to its devices on the bus to which you are connecting the controllers.

2. Refer to Figure 4–6 for the balance of these steps: Remove the trilink connectors (① 12–39921–01) from both controllers. These connectors are 68-pin Y-adapters that maintain bus continuity even when they are disconnected from their controller.

3. Connect a bus cable ② from the host to one of the connectors on the front of each trilink connector.

4. If you are connecting a host to a controller in a BA370 enclosure that will reside in an SW600 cabinet, snap a ferrite bead ③ on each bus cable within one inch of the controller.

   If you are connecting a host to a controller in any other enclosure or cabinet, skip to step 5.

Figure 4–6   Connecting Multiple Bus Failover, Dual-Redundant Controllers to the Host
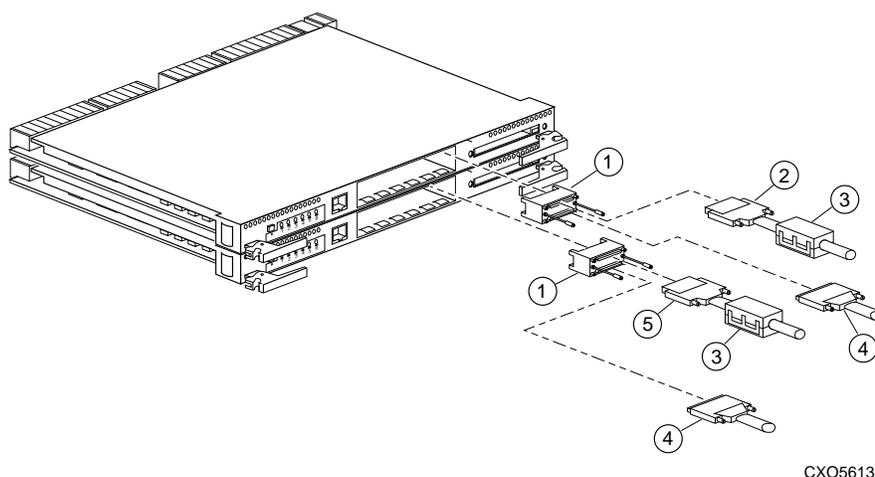
CXO5613B

5. Perform one of the following:

■ If the controllers are at the end of the bus, connect a terminator ④ to the open connectors on the front of each trilink connector.

■ If the controllers are not at the end of the bus, connect a cable ⑤ that continues to the next device on each bus and install the terminator at the end of the bus.

6. Reconnect the trilink connectors to host ports on the controllers. Do not disconnect the host cables from the trilink connectors.

7. Route and tie the cables as desired.

8. Restart the I/O from the host. Some operating systems may require you to restart the host to see the devices attached to the new controller.

Multiple-Bus Failover Mode CLI Configuration Procedure

1. Apply power to the subsystem.

   The powerup sequence takes approximately 45 seconds. At the end of the powerup sequence, the audible alarm on the EMU will sound and the EMU error LED will be solidly lit. Turn off the alarm by pressing and releasing the EMU reset button once.

Note   The audible alarm will cease, but the LED remains on until the controllers are bound into failover mode. The CLI will display a copyright notice and a report for the "other" controller.

---

2.   Ensure the Power Verification and Addressing (PVA) module ID switch(es) in the array subsystem are set to the following:

■   Controller enclosure—set the PVA switch in that cabinet to 0.

■   The next enclosure in an extended controller subsystem—set the PVA switch in that cabinet to 2.

3.   The third cabinet in an extended controller subsystem—set the PVA switch in that cabinet to 3.

4.   Enter the following command to stop the CLI from reporting a misconfiguration error resulting from having no failover mode specified:

CLEAR CLI

5.   Ensure that neither of the controllers is already placed into a failover mode by using the SET NOFAILOVER command:

SET NOFAILOVER

6.   Place the controller pair into a multiple-bus failover mode by using the SET MULTIBUS_FAILOVER COPY= *controller* command:

SET MULTIBUS_FAILOVER COPY=THIS_CONTROLLER

The copy qualifier normally specifies where the good copy of the array configuration is. At this point in the configuration process there is no array configuration, but the qualifier must be specified as it is part of the command syntax.

Note   When the command is entered, both controllers restart. The restart may set off the EMU audible alarm. To silence the alarm, press the EMU alarm button once and release. The audible alarm will cease, but the LED remains on until the controllers bind into failover mode. This process takes about 15 seconds.

The CLI will print out a report from the "other" controller indicating the "other" controller restarted. The CLI will continue reporting this condition until cleared:

CLEAR CLI

7. Set up mirrored cache (if desired) for the controller pair with the SET *controller* with the *MIRRORED_CACHE* switch:

SET THIS_CONTROLLER MIRRORED_CACHE
SET OTHER_CONTROLLER MIRRORED_CACHE

Note   This command causes a restart, so the EMU audible alarm may sound.

---

8. Assign controller targets to the host port according to the planned configuration. For example, to assign targets 1 and 2 use the SET *controller* with the *ID=* switch:

SET THIS_CONTROLLER ID=1,2

This command requires both controllers to restart before taking effect.

9. Restart the "other" controller first, then "this" controller with the RESTART *controller* command:

RESTART OTHER_CONTROLLER
RESTART THIS_CONTROLLER

Note   When the "other" controller restarts, the EMU audible alarm sounds, but ceases when "this" controller restarts.

It takes about one minute for the CLI prompt to come back after the issuance of a RESTART command.

10. If desired, prefer targets to "this" controller. As an example, to prefer target 2 to "this" controller, use the SET *controller* with the *PREFERRED_ID* switch:

SET THIS_CONTROLLER PREFERRED_ID=2

11. Set the time on the controller with the SET *controller* with the *TIME=* switch:

SET THIS_CONTROLLER TIME= DD-MMM-YYYY:HH:MM:SS

12. Use the FRUTIL utility to set up the battery discharge timer:

RUN FRUTIL

Note   The RESTART THIS_CONTROLLER command begins a series of memory tests which take a few minutes to run. If the diagnostics are still running, FRUTIL will not start and a message informs you that memory tests are running. FRUTIL starts after the tests complete.

Enter "Y" to the following FRUTIL display:
```
Do you intend to replace this controller's cache
battery? Y/N [N]
```

FRUTIL prints out a procedure, but won't give you a prompt. Ignore the procedure and press Enter.

13. Move the maintenance cable to the "other" controller, and repeat steps 8 and 9.

14. Set up any additional optional controller settings (for example, changing the CLI prompt). See the SET *this_controller* (SET *other_controller*) command in the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for the format of the options.

15. Enter a SHOW t*his_controller* and a SHOW *other-controller* command to verify that all changes have taken place:

SHOW THIS_CONTROLLER
SHOW OTHER_CONTROLLER

Configuring Devices

The devices on the device bus can be configured either manually or by the CONFIG utility. The CONFIG utility is easier to use.

1. Invoke the CONFIG utility by using the RUN *program-name* command:

RUN CONFIG

CONFIG takes about two minutes to locate and map the configuration of a completely populated subsystem. CONFIG displays a message that indicates it is running that is similar to the following example:

```
Config Local Program Invoked

Config is building its tables and determining what
devices exist on the subsystem. Please be patient.

 add disk        DISK10000        1 0 0

 add disk        DISK10100        1 1 0

 add disk        DISK20000        2 0 0

 add disk        DISK20100        2 1 0

 . . . .


Config - Normal Termination
```

2. Initialize each device separately using the INITIALIZE *container-name* command:

    INITIALIZE DISK10000
    INITIALIZE DISK10100
    INITIALIZE DISK20000
    INITIALIZE DISK20100
    . . . . . .

Configuring a Stripeset

Use the following procedure to configure a stripeset:

1. Create the stripeset by adding its name to the controller's list of storagesets and specifying the disk drives it contains by using the ADD STRIPESET *stripeset-name* c*ontainer-name1 container-name2 container-name n* command:

    ADD STRIPESET STRIPESET-NAME DISKNNNNN DISKNNNNN

2. Initialize the stripeset. If you want to set any INITIALIZE switches, you must do so in this step:

    INITIALIZE STRIPESET-NAME SWITCH

3. Verify the stripeset configuration and switches with the SHOW *stripeset-name* command:

    SHOW STRIPESET-NAME

4. Assign the stripeset a unit number to make it accessible by the host or hosts (see "ADD UNIT D300 MIRROR1 PARTITION=3," page 4-25).

The following is an example showing the commands needed to create STRIPE1, a 3-member stripeset:

ADD STRIPESET STRIPE1 DISK1000 DISK2000 DISK3000
INITIALIZE STRIPE1 CHUNKSIZE=128
SHOW STRIPE1

Note   Refer to Chapter 3 for detailed information on stripeset switches and values.

Configuring a Mirrorset

Use the following procedure to configure a mirrorset:

1. Create the mirrorset by adding the mirrorset name to the controller's list of storagesets and specifying the disk drives it contains. Optionally, you can append Mirrorset switch values. If you do not specify switch values the default values are applied. Use the ADD MIRRORSET *mirrorset-name disk-name1 disk-name2 disk-name n* command:

ADD MIRRORSET MIRRORSET-NAME DISKNNNNN
DISKNNNNN SWITCH

2. Initialize the mirrorset. if you want to set any INITIALIZE switches, you must do so in this step:

INITIALIZE MIRRORSET-NAME SWITCH

3. Verify the mirrorset configuration and switches by using the SHOW *mirrorset-name* command:

SHOW MIRRORSET-NAME

4. Assign the mirrorset a unit number to make it accessible by the host or hosts (see "ADD UNIT D300 MIRROR1 PARTITION=3," page 4-25).

The following is an example of the commands needed to create MIRR1, a 2-member mirrorset:

ADD MIRRORSET MIRR1 DISK10000 DISK20000
INITIALIZE MIRR1
SHOW MIRR1

Refer to Chapter 3 for detailed information on mirrorset switches and values.

Configuring a RAIDset

Use the following procedure to configure a RAIDset:

1. Create the RAIDset by adding its name to the controllers list of storagesets and specifying the disk drives it contains. Optionally, you can append RAIDset switch values. If you do not specify switch values, the default values are applied. Use the ADD RAIDSET *RAIDset-name container-name1 container-name2 container-name n* command:

    ADD RAIDSET RAIDSET-NAME DISKNNNN DISKNNNN DISKNNNN SWITCH

2. Initialize the RAIDset. If you want to set any optional INITIALIZE switches, you must do so in this step):

    INITIALIZE RAIDSET-NAME SWITCH

    Note   It is recommended that you allow initial reconstruct to complete before allowing I/O to the RAIDset. Not doing so may generate forced errors at the host level. To determine whether initial reconstruct has completed, enter SHOW RAIDSET *FULL*.

    ---

3. Verify the RAIDset configuration and switches by using the SHOW *raidset-name* command:

    SHOW RAIDSET-NAME

4. Assign the RAIDset a unit number to make it accessible by the host or hosts (see "ADD UNIT D300 MIRROR1 PARTITION=3," page 4-25).

    The following is an example of the commands needed to create RAID1, a 3-member RAIDset:

    **ADD RAIDSET RAID1 DISK10000 DISK20000 DISK3000
    INITIALIZE RAID1
    SHOW RAID1**

    See Chapter 3 for detailed information on RAIDset switches and values.

Configuring a Striped Mirrorset

Use the following procedure to configure a striped mirrorset:

1. Create two or more mirrorsets using the ADD MIRRORSET *mirrorset-name disk-name1 disk-name2 disk-name n* command (do not initialize):

   ADD MIRRORSET MIRRORSET-NAME1 DISKNNNN DISKNNNN
   ADD MIRRORSET MIRRORSET-NAME2 DISKNNNN DISKNNNN

2. Create a stripeset using the ADD STRIPESET *stripeset-name container-name1 container-name2 container name n* specifying the name of the stripeset and the names of the newly-created mirrorset(s) from step1:

   ADD STRIPESET STRIPESET-NAME MIRRORSET-NAME1
   MIRRORSET-NAME2

3. Initialize the stripeset. If you want to set any optional INITIALIZE switches, you must do so in this step):

   INITIALIZE STRIPESET-NAME SWITCH

4. Verify the striped mirrorset configuration and switches:

   SHOW STRIPESET-NAME

5. Assign the striped mirrorset a unit number to make it accessible by the host or hosts (see Chapter 4, "ADD UNIT D300 MIRROR1 PARTITION=3").

   The following is an example of the commands needed to create a striped mirrorset with the name of Stripe1. Stripe1 is a 3-member striped mirrorset made up of Mirr1, Mirr2, and Mirr3 (each of which is a 2-member mirrorset):

   ADD MIRRORSET MIRR1 DISK10000 DISK20000
   ADD MIRRORSET MIRR2 DISK30000 DISK40000
   ADD MIRRORSET MIRR3 DISK50000 DISK60000
   ADD STRIPESET STRIPE1 MIRR1 MIRR2 MIRR3
   INITIALIZE STRIPE1 CHUNKSIZE=DEFAULT
   SHOW STRIPE1

   See Chapter 3 for detailed information on stripeset and mirrorset switches and values.

Configuring a Single Disk Unit

Follow this procedure to set up a single disk unit in your subsystem:

1. Initialize the disk drive, and set up the desired switches for the disk with the INITIALIZE *container-name SWITCH* command:

    INITIALIZE DISKNNN SWITCH

2. Assign the disk a unit number to make it accessible by the host or hosts (see Chapter 4, "Assigning a Unit Number to a Single (JBOD) Disk").

3. Verify the configuration:

    SHOW DEVICE

Configuring a Partition

A single disk drive may be partitioned as well as any of the storagesets. Use one of the following two procedures to configure a partition from a single disk drive or a storageset.

Note   Partitions cannot be used for a dual-redundant controller pair operating in multiple-bus failover mode.

Partitioning a Storageset

1. Add the storageset (and its name) to the controller's list of storagesets and specify the disk drives it contains by using the appropriate ADD command:

    ADD STORAGESET-TYPE STORAGESET-NAME DISKNNNN DISKNNNN

2. Initialize the storageset. If you want to set any INITIALIZE switches, you must do so in this step:

    INITIALIZE STORAGESET-NAME SWITCH

3. Create each partition in the storageset by using the CREATE_PARTITION *container-name SIZE=* command:

    CREATE_PARTITION STORAGESET-NAME SIZE=N

    where *SIZE=N* may be a percentage of the storageset that will be assigned to the partition. Enter *SIZE=LARGEST* to let the controller assign the largest free space available to the partition.

4. Verify the partitions:

SHOW STORAGESET-NAME

The partition number appears in the first column, followed by the size and starting block of each partition.

5. Assign the partitioned storageset a unit number to make it accessible by the host(s). Refer to Chapter 4, "Preferring Units in Multiple-Bus Failover Mode".

The following is an example of the commands needed to create a partitioned RAIDset named RAID1. RAID1 is a 3-member RAIDset, partitioned into two storage units:

ADD RAIDSET RAID1 DISK10000 DISK20000 DISK30000
INTIALIZE RAID1
CREATE_PARTITION RAID1 SIZE=25
CREATE_PARTITION RAID1 SIZE=LARGEST
SHOW RAID1

See Chapter 3 for detailed information on switches and values.

Partitioning a Single Disk Drive

1. Add the disk drive to the controller's list of containers by using the ADD DISK *container-name SCSI-port-location* command:

    ADD DISK DISKNNNN PTL-LOCATION

2. Initialize the storageset or disk drive. If you want to set any INITIALIZE switches, you must do so in this step:

    INITIALIZE CONTAINER-NAME SWITCH

3. Create each partition in the disk drive by using the CREATE_PARTITION *container-name SIZE=* command:

    CREATE_PARTITION STORAGESET-NAME SIZE=N

    where *SIZE=N* may be a percentage of the disk drive that will be assigned to the partition. Enter *SIZE=LARGEST* to let the controller assign the largest free space available to the partition.

4. Verify the partitions:

    SHOW STORAGESET-NAME

    The partition number appears in the first column, followed by the size and starting block of each partition.

5. Assign the disk a unit number to make it accessible by the host(s). Refer to Chapter 4, "Preferring Units in Multiple-Bus Failover Mode".

The following is an example of the commands needed to create DISK1 partitioned into three storage units:

```
ADD DISK DISK1 10200
INTIALIZE DISK1
CREATE_PARTITION DISK1 SIZE=25
CREATE_PARTITION DISK1 SIZE=25
CREATE_PARTITION DISK1 SIZE=50
SHOW DISK1
```

See Chapter 3 for detailed information on partition switches and values.

Assigning Unit Numbers and Unit Qualifiers

Each storageset, partition, or single (JBOD) disk must be assigned a unit number for the host to access. As the units are added, their properties can be specified through the use of command switches, which are described in detail under the ADD UNIT command in the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual*.

The ADD UNIT command gives a storageset a logical unit number by which the host(s) can access it. The unit umber must be associated with one of the target IDs on the host bus.

Assigning a Unit Number to a Partition

Use the ADD UNIT *unit-number container-name PARTITION=* command to assign a unit number to a storageset partition:

```
ADD UNIT UNIT_NUMBER STORAGESET_NAME
PARTITION=PARTITION-NUMBER
```

The following is an example of the command syntax needed to assign partition 3 of mirrorset mirr1 to target 3 LUN 0:

```
ADD UNIT D300 MIRROR1 PARTITION=3
```

Assigning a Unit Number to a Storageset

Use the ADD UNIT *unit-number container-name* command to assign a logical unit number to a storageset:

```
ADD UNIT UNIT_NUMBER STORAGESET_NAME
```

The following is an example of the command syntax needed to assign storageset R1 to target 1 LUN 2:

```
ADD UNIT D102 R1
```

Assigning a Unit Number to a Single (JBOD) Disk

Use the ADD UNIT *unit-number container-name* command to assign a unit number to a single disk:

ADD UNIT UNIT_NUMBER DISK_NAME

The following is an example of the command syntax needed to assign disk 20300 to target 0 LUN 42:

ADD UNIT D4 DISK20300

Preferring Units in Multiple-Bus Failover Mode

In multiple-bus failover mode, individual units can be preferred (assigned) to a specific controller.

The SET *controller PREFERRED_PATH=* command is used to establish the preferring (assigning) of units to a particular controller.

The following is an example of the command syntax needed to prefer unit D102 to THIS_CONTROLLER:

SET D102 PREFERRED_PATH=THIS_c0NTROLLER

Configuration Options

During the configuration process, there are many options to choose from. This section describes how to set up some of the more common ones.

Changing the CLI Prompt

To change the CLI prompt from the default, enter a 1—16 character new prompt string in the switch field of the SET *controller PROMPT=* command:

SET THIS_CONTROLLER PROMPT = NEW PROMPT

If you are configuring dual-redundant controllers, change the CLI prompt on the "other" controller as well:

SET OTHER_CONTROLLER PROMPT = NEW PROMPT

Setting Maximum Data Transfer Rate

To set the maximum data transfer rate from the default, use the SET *controller TRANSFER_RATE_REQUESTED=* command:

SET THIS_CONTROLLER TRANSFER_RATE_REQUESTED = SPEED

If you are configuring dual-redundant controllers, change the maximum data transfer rate on the "other" controller as well:

SET OTHER_CONTROLLER TRANSFER_RATE_REQUESTED = SPEED

Set-Up Cache UPS

By default, the controller expects to use an external cache battery (ECB) as the power source backup to the cache module. You can instead choose to use an uninterruptable power supply (UPS) to provide this backup in the event of a primary power failure.

To support your subsystem with an UPS, use the SET *controller CACHE_UPS* command:

SET THIS_CONTROLLER CACHE_UPS

Note   The companion controller in a dual-redundant pair inherits the cache UPS setting.

Adding Disk Drives

Any factory-installed devices in your StorageWorks subsystem have already been added to the controller's list of eligible storage devices. if new drives are to be added to your subsystem, you must issue one of the CLI commands found in the following paragraphs before you can use them in any kind of storageset, single disk unit, or spareset:

- Adding One Disk Drive at a Time
- Adding Several Disk Drives at a Time
- Adding a Disk Drive to the Spareset

Adding One Disk Drive at a Time

To add one new disk drive to your controller's list of eligible storage devices, use the ADD DISK *container-name SCSI-port-location* command:

ADD DISK DISKNNNN PTL-LOCATION SWITCH_VALUE

Adding Several Disk Drives at a Time

To add several new disk drives to your controller's list of eligible storage devices, use the RUN *program-name* command to start the CONFIG utility:

RUN CONFIG

Adding/Deleting a Disk Drive to the Spareset

The spareset is a collection of hot spares that are available to the controller should it need to replace a failed member of a RAIDSET or mirrorset. The following procedures describe how to add and delete a disk drive to/from the spareset.

Use the following steps to add a disk drive to the spareset:

1. Add the disk drive to the controller's list of containers by using the ADD DISK *container-name SCSI-port-location* command:

   ADD DISK DISKNNNN PTL-LOCATION.

2. Add the disk drive to the spareset list by using the ADD SPARESET *disk-name* command:

   ADD SPARESET DISK NNNN

   Repeat this step for each disk drive you want added to the spareset.

3. Verify the contents of the spareset:

   SHOW SPARESET

   The following is an example of the command syntax needed to add DISK60000 and DISK60100 to the spareset:

   ADD SPARESET DISK60000
   ADD SPARESET DISK60100
   SHOW SPARESET

   You cannot delete the spareset—it always exists whether or not it contains disk drives. You can, however, delete disks within the spareset if you need to use them elsewhere in your subsystem.

To remove a disk drive from the spareset:

1. Show the contents of the spareset with the following command:

   SHOW SPARESET

2. Delete the desired disk drive by using the DELETE SPARESET *disk-name* command:

   DELETE SPARESET DISKNNNN

3. Verify the contents of the spareset:

   SHOW SPARESET

   The following is an example of the command syntax needed to remove DISK60000 from the spareset:

   SHOW SPARESET

Name	Storageset	Uses	Used By
SPARESET	spareset	DISK60000	
		DISK60100	

   DELETE SPARESET DISK60000
   SHOW SPARESET

Name	Storageset	Uses	Used By
SPARESET	spareset	DISK60100	

Enabling/Disabling Autospare

The autospare feature allows any new disk drive that is inserted into the PTL location of a failed drive to be automatically initialized and placed into the spareset.

To enable autospare, use the SET FAILEDSET *replacement policy* command using *autospare* as the parameter:

SET FAILEDSET AUTOSPARE

To disable autospare, use the following command:

SET FAILEDSET NOAUTOSPARE

During initialization, the *autospare* parameter of the SET FAILEDSET command checks for metadata on the new disk drive. Metadata is the information that indicates the drive belongs to, or has been used by, a known storageset. If the drive contains metadata, initialization stops.

Note   A new disk drive will not contain metadata, but a repaired or re-used disk drive may contain metadata.

To erase metadata from a disk drive:

1. Add the disk drive to the controller's list of eligible devices by using the ADD DISK *container-name SCSI-port-location* command:

    ADD DISK DISKNNNN PTL-LOCATION

2. Make the device transportable by using the SET *device-name TRANSPORTABLE* command:

    SET DISKNNNN TRANSPORTABLE

3. Initialize the device by using the INITIALIZE *container-name* command:

    INITIALIZE DISKNNNN

Deleting a Storageset

Any storageset may be deleted unless it is partitioned. If the storageset you want to delete is partitioned, you must first delete each partitioned unit before you can delete the storageset. Use the following steps to delete a storageset:

1. Show the configuration:

    SHOW STORAGESETS

2. Delete the unit number shown in the "Used by" column:

    DELETE UNIT-NUMBER

3. Delete the name shown in the "Name" column:

    DELETE STORAGESET-NAME

4. Verify the configuration:

    SHOW STORAGESETS

The following is an example of the command syntax needed to delete Stripe1, a 3-member stripeset that is made up of DISK10000, DISK20000, and DISK30000:

SHOW STORAGESETS

Name	Storageset	Uses	Used By
STRIPE1	stripeset	DISK10000	D100
		DISK20000	
		DISK30000	

DELETE D100
DELETE STRIPE1
SHOW STORAGESETS

Changing Switches for a Storageset or Device

You can optimize a storageset or device at any time by changing the switches that are associated with it. Remember to update the storageset profile when you change its switch configuration. This section describes the following:

■ Displaying the Current Switches

■ Changing RAIDset and Mirrorset Switches

■ Changing Device Switches

■ Changing Initialize Switches

■ Changing Unit Switches

Displaying the Current Switches

To display the current switches for a storageset or single-disk unit, enter the following CLI command:

SHOW STORAGESET-NAME FULL

or

SHOW DEVICE-NAME FULL

Changing RAIDset and Mirrorset Switches

Use the SET RAIDset-name or SET mirrorset-name command to change the RAIDset or mirrorset switches associated with an existing storageset.

For example, the following command changes the replacement policy for RAIDset RAID1 to *BEST_FIT*:

SET RAID1 POLICY=BEST_FIT

Changing Device Switches

Use the SET *unit-number* command to change the device switches. For example, the following command enables DISK10000 to be used in a non-StorageWorks environment:

SET DISK1000 TRANSPORTABLE

The *TRANSPORTABLE* switch cannot be changed for a disk if the disk is part of an higher-level container. Additionally, the disk cannot be configured as a unit if it is to be used as indicated in the preceding example.

Changing Initialize Switches

The INITIALIZE switches cannot be changed without destroying the data on the storageset or device. These switches are integral to the formatting and can only be changed by reinitializing the storageset.

> ⚠️ **Caution**   Initializing a storageset is similar to formatting a disk drive; all data is destroyed during this procedure.

Changing Unit Switches

Use the SET *unit-number* command to change unit switches that are associated with a unit. For example, the following CLI command enables write protection for unit D100:

SET D100 WRITE_PROTECT

C H A P T E R 5

Periodic Procedures

This chapter describes procedures you might need when working with your HSZ70 array controller:

Formatting Disk Drives

Use the HSUTIL FORMAT option to simultaneously format up to seven disk drives attached to a single controller or up to six disk drives attached to a dual-redundant pair of controllers.

Consider the following points before formatting disk drives with HSUTIL:

■   HSUTIL can format any unconfigured disk drive. A configured device may be formatted only:

–   After its unit number is deleted and

–   The device is removed from its storageset (RAIDset, Stripeset, Mirrorset, Failedset, or Spareset).

Refer to the CLI DELETE command in the *HSZ70 Array Controller HSOF Version 7.3 CLI Reference Manual* for further information on deleting units and storagesets.

■   If the power fails or the bus is reset while HSUTIL is formatting a disk drive, the drive may become unusable. To minimize this possibility, DIGITAL recommends you use a reliable power source and suspend all non-HSUTIL activity to the bus that services the target disk drive.

■   HSUTIL cannot control or affect the defect management for a disk drive.

■   Do not invoke any CLI command or run any local program that might reference the target disk drive while HSUTIL is active. Also, do not reinitialize either controller in the dual-redundant configuration while using HSUTIL.

Caution   Suspend all I/O to the buses that service the target disk drives before using HSUTIL to format the drives.

Using the HSUTIL Utility Program

To format one or more disk drives complete the following steps:

1.   Start HSUTIL by issuing the following command:

     RUN HSUTIL

2.   Enter 1 to select the FORMAT function.

     HSUTIL finds and displays all of the unattached disk drives configured on the controller.

3.   Type the name of a disk drive you want to format.

4.   Enter Y to enter another disk drive name or N to begin the formatting operation.

5.   Read the cautionary information that HSUTIL displays, then confirm or cancel the formatting operation.

     The formatting operation will complete in approximately the estimated time.

Example 5–1

The following example shows the sequence of steps as they would appear on the terminal when you format a disk drive:

```
CLI> RUN HSUTIL

*** AVAILABLE FUNCTIONS ARE:
    0. EXIT
    1. FORMAT
    2. DEVICE_CODE_LOAD_DISK
    3. DEVICE_CODE_LOAD_TAPE
    ENTER FUNCTION NUMBER (0:3) [0] ? 1

UNATTACHED DEVICES ON THIS CONTROLLER INCLUDE:
    DEVICE          SCSI PRODUCT ID    CURRENT DEVICE REV
    DISK10000        RZ26   (C) DEC     T386
    DISK20000        RZ26   (C) DEC     T386
    DISK20100        RZ29B  (C) DEC     0006
    DISK30100        RZ25   (C) DEC     0900
    DISK30200        RZ26L  (C) DEC     X442

ENTER A DEVICE TO FORMAT ? DISK10000
    FORMAT DISK10000 MAY TAKE UP TO 40 MINUTES TO FORMAT
    SELECT ANOTHER DEVICE (Y/N) [N] Y

ENTER A DEVICE TO FORMAT ? DISK20000
    FORMAT DISK20000 MAY TAKE UP TO 35 MINUTES TO FORMAT
    SELECT ANOTHER DEVICE (Y/N) [N] Y

ENTER A DEVICE TO FORMAT ? DISK20100
    FORMAT DISK20100 MAY TAKE UP TO 15 MINUTES TO FORMAT
    SELECT ANOTHER DEVICE (Y/N) [N] N

^Y AND ^C WILL BE DISABLED WHILE THE FORMAT OPERATION IS IN
    PROGRESS.

CAUTION:
      WHEN YOU FORMAT A DEVICE, IT WILL DESTROY THE DATA ON
      THE DEVICE. A BACKUP OF THE DEVICE SHOULD HAVE BEEN
      DONE IF THE DATA IS IMPORTANT.

NOTE:
      IN ORDER TO MINIMIZE THE POSSIBILITY OF A SCSI BUS
      RESET, IT IS RECOMMENDED THAT YOU PREVENT NON-HSUTIL IO
      OPERATIONS TO ALL OTHER DEVICES ON THE SAME PORT AS THE
      DESTINATION DEVICE(S). IF A SCSI BUS RESET OCCURS, THE
```

```
       FORMAT MAY BE INCOMPLETE AND YOU MAY HAVE TO RE-INVOKE
       HSUTIL.

AFTER YOU ANSWER THE NEXT QUESTION, THE FORMAT WILL START. DO YOU
   WANT TO CONTINUE (Y/N) [N] ? Y

HSUTIL STARTED AT: 14-JAN-1997 15:00:31
    FORMAT OF DISK10000 FINISHED AT 14-JAN-1997 15:25:12
    FORMAT OF DISK20000 FINISHED AT 14-JAN-1997 15:30:31
    FORMAT OF DISK20100 FINISHED AT 14-JAN-1997 15:30:43

HSUTIL - NORMAL TERMINATION AT 14-JAN-1997 15:31:09
```

Clone Utility

Use the CLONE utility to duplicate the data on any unpartitioned single-disk unit, stripeset, mirrorset, or striped mirrorset in preparation for backup. When the cloning operation is done, you can back up the clones rather than the storageset or single-disk unit, which can continue to service its I/O load. When you are cloning a mirrorset, CLONE does not need to create a temporary mirrorset. Instead, it adds a temporary member to the mirrorset and copies the data onto this new member.

The CLONE utility creates a temporary, two-member mirrorset for each member in a single-disk unit or stripeset. Each temporary mirrorset contains one disk drive from the unit you are cloning and one disk drive onto which CLONE copies the data. During the copy operation, the unit remains online and active so the clones contain the most up-to-date data.

After the CLONE utility copies the data from the members to the clones, it restores the unit to its original configuration and creates a clone unit you can backup. The CLONE utility uses steps shown in Figure 5–1 to duplicate each member of a unit.

Figure 5–1   CLONE Steps for Duplicating Unit Members
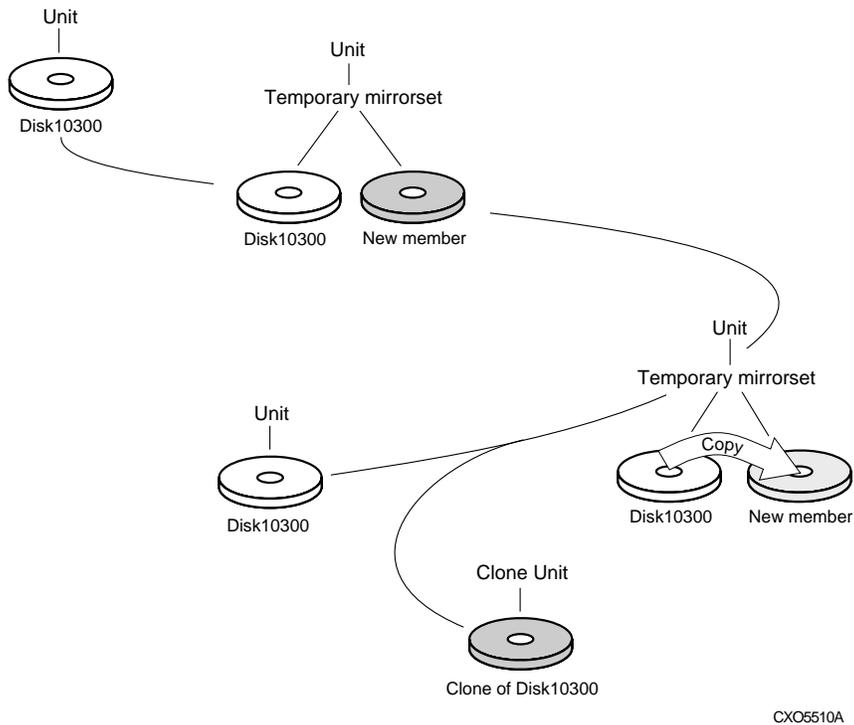
Cloning a Single-Disk Unit, Stripeset, or Mirrorset

Use the following procedure to clone a single-disk unit, stripeset, or mirrorset:

1. Establish a connection to the controller that accesses the unit you want to clone.

2. Start CLONE using the following syntax:

   RUN CLONE

3. When prompted, enter the unit number of the unit you want to clone.

4. When prompted, enter a unit number for the clone unit that CLONE will create.

5. When prompted, indicate how you would like the clone unit to be brought online: either automatically or only after your approval.

6.   When prompted, enter the disk drives you want to use for the clone units.

7.   Back up the clone unit.

Example 5–1
This example shows the commands you would use to clone storage unit D204. The clone command terminates after it creates storage unit D205, a clone or copy of D204.

RUN CLONE

```
CLONE LOCAL PROGRAM INVOKED

UNITS AVAILABLE FOR CLONING:101

204
```

ENTER UNIT TO CLONE ? 204

```
CLONE WILL CREATE A NEW UNIT WHICH IS A COPY OF UNIT 204.
```

ENTER THE UNIT NUMBER WHICH YOU WANT ASSIGNED TO THE NEW UNIT ? 205

```
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:

DISK20200 (SIZE=832317)

DISK20400 (SIZE=832317)

DISK30100 (SIZE=832317)
```

USE AVAILABLE DEVICE DISK20200(SIZE=832317) FOR MEMBER DISK10300(SIZE=832317) (Y,N) [Y] ? Y

```
MIRROR DISK10300 C_MA

SET C_MA NOPOLICY
```

```
SET C_MA MEMBERS=2

SET C_MA REPLACE=DISK220

DEVICES AVAILABLE FOR CLONE TARGETS:

DISK20400 (SIZE=832317)

DISK30100 (SIZE=832317)
```

USE AVAILABLE DEVICE DISK20400(SIZE=832317) FOR MEMBER DISK20000(SIZE=832317) (Y,N) [Y] ? Y

```
MIRROR DISK20000 C_MB

SET C_MB NOPOLICY

SET C_MB MEMBERS=2

SET C_MB REPLACE=DISK20400

COPY IN PROGRESS FOR EACH NEW MEMBER. PLEASE BE PATIENT...

.

.

COPY FROM DISK10300 TO DISK20200 IS 100% COMPLETE

COPY FROM DISK20000 TO DISK20400 IS 100% COMPLETE
```

PRESS RETURN WHEN YOU WANT THE NEW UNIT TO BE CREATED

```
REDUCE   DISK20200 DISK20400

UNMIRROR DISK10300

UNMIRROR DISK20000

ADD MIRRORSET C_MA       DISK20200

ADD MIRRORSET C_MB       DISK20400

ADD STRIPESET C_ST1  C_MA C_MB

INIT C_ST1     NODESTROY CHUNK=128

ADD UNIT D205 C_ST1

D205 HAS BEEN CREATED. IT IS A CLONE OF D204.

CLONE - NORMAL TERMINATION
```

Backing Up Your Subsystem Configuration

Your controller stores information about your subsystem configuration in nonvolatile memory (NVMEM). This information could be lost if the controller fails or when you replace a module in your subsystem.

You can avoid reconfiguring your subsystem manually by saving configuration information on one or more of your subsystem disks using the INITIALIZE *SAVE_CONFIGURATION* command. The controller updates the configuration information saved to disk whenever it changes. If the controller fails or you replace a module, you can easily restore your subsystem configuration from this information on the disks. Storing the configuration information uses a small amount of space on each device.

You do not need to store the configuration on all devices in the subsystem. You can use the INITIALIZE command without the *SAVE_CONFIGURATION* switch option for any devices on which you do not want to save the configuration.

You cannot use the *SAVE_CONFIGURATION* switch on TRANSPORTABLE disks.

Saving Subsystem Configuration Information to a Single Disk

You can choose to save your subsystem configuration information on a single disk.

Choose a disk on which to save the information by using the *SAVE_CONFIGURATION* switch when you initialize the disk with the INITIALIZE command. Use the following syntax:

INITIALIZE DISK*nnn* **SAVE_CONFIGURATION**

Saving Subsystem Configuration Information to Multiple Disks

You can save your subsystem configuration information to as many individual disks as you would like, but you must initialize each using the *SAVE_CONFIGURATION* switch. Use the following syntax for each:

INITIALIZE DISK*nnn* **SAVE_CONFIGURATION**

Saving Subsystem Configuration Information to a Storageset

You can save your subsystem configuration information to a storageset. The configuration information is duplicated on every disk that is a member of the storageset. Use the following syntax:

INITIALIZE *storageset-name* **SAVE_CONFIGURATION**

Displaying the Status of the Save Configuration Feature

You can use the SHOW THIS_CONTROLLER FULL command to find out if the save configuration feature is active and which devices are being used to store the configuration. The display includes a line indicating status and how many devices have copies of the configuration, as shown in the following example.

Example 5–2

SHOW THIS_CONTROLLER FULL

```
CONTROLLER:

      HSZ70    (C) DEC ZG64100138 FIRMWARE QBFB-0, HARDWARE CX02
         CONFIGURED FOR DUAL-REDUNDANCY WITH ZG64100209
             IN DUAL-REDUNDANT CONFIGURATION
         DEVICE PORT SCSI ADDRESS 7
         TIME: NOT SET
   HOST PORT:
         SCSI TARGET(S) (1, 3, 11)
         PREFERRED TARGET(S) (3, 11)
         TRANSFER_RATE_REQUESTED = 20MHZ
         HOST FUNCTIONALITY MODE = A
         COMMAND CONSOLE LUN IS TARGET 1, LUN 5
   CACHE:
         64 MEGABYTE WRITE CACHE, VERSION 4
         CACHE IS GOOD
         BATTERY IS GOOD
         NO UNFLUSHED DATA IN CACHE
         CACHE_FLUSH_TIMER = DEFAULT (10 SECONDS)
         NOCACHE_UPS
   MIRRORED CACHE:
         64 MEGABYTE WRITE CACHE, VERSION 4
         CACHE IS GOOD
         BATTERY IS GOOD
         NO UNFLUSHED DATA IN CACHE
```

```
EXTENDED INFORMATION:
        TERMINAL SPEED 19200 BAUD, EIGHT BIT, NO PARITY, 1 STOP
BIT
        OPERATION CONTROL: 00000001  SECURITY STATE CODE: 75524
       CONFIGURATION BACKUP ENABLED ON 3 DEVICES
```

Refer to the following example for sample devices with the
SAVE_CONFIGURATION switch enabled.

Example 5–3

$ SHOW DEVICES FULL

```
NAME            TYPE                       PORT TARG  LUN      USED BY


----------------------------------------------------------------
   DISK10000     DISK                        1    0    0        S2
            DEC     RZ28M    (C) DEC 1003

      SWITCHES:
            NOTRANSPORTABLE
            TRANSFER_RATE_REQUESTED = 20MHZ (SYNCHRONOUS 10.00
   MHZ NEGOTIATED)
         SIZE: 4108970 BLOCKS
         CONFIGURATION BEING BACKED UP ON THIS CONTAINER
   DISK30300     DISK                        3    3    0        S2
            DEC     RZ28M    (C) DEC 1003
          SWITCHES:
            NOTRANSPORTABLE
            TRANSFER_RATE_REQUESTED = 20MHZ (SYNCHRONOUS 10.00
   MHZ NEGOTIATED)

      SIZE: 4108970 BLOCKS

      CONFIGURATION BEING BACKED UP ON THIS CONTAINER

 TAPE40300     PASSTHROUGH TAPE              4    3    0       P1107

       DEC     TZS20    (C)DEC 01AB

      SWITCHES:

 TRANSFER_RATE_REQUESTED = 20MHZ (SYNCHRONOUS 10.00 MHZ
   NEGOTIATED)
```

Shutting Down Your Subsystem

Follow these steps to shut down your StorageWorks subsystem:

1. On the host, dismount the storage units in your subsystem.

2. Connect a maintenance terminal to one of the controllers in your subsystem.

3. Shut down the controllers. If you have dual-redundant controllers, shut down the "other controller" first, then shut down "this controller." Use the following syntax:

    SHUTDOWN OTHER_CONTROLLER
    SHUTDOWN THIS_CONTROLLER

    Note   This process can take up to five minutes to complete depending on the amount of data to be flushed from cache.

    ---

4. Turn off the power to the subsystem.

5. Unplug the subsystem power cord.

6. Disable the ECB by pressing its shut off button until its status light stops blinking—about two seconds.

Restarting Your Subsystem

Follow these steps to restart your subsystem:

1. Plug in the subsystem power cord.

2. Turn on the subsystem.

3. Press and hold the controller reset button for 3 seconds (then release).

4. Check the status of the write-back cache module backup battery. If your subsystem has been off for an extended period of time, the battery may be drained. Use the following syntax to check the battery status:

    SHOW THIS_CONTROLLER

APPENDIX A

Controller Specifications

This appendix contains physical, electrical, and environmental specifications for the HSZ70 array controller:

■ "Controller Specifications," page A-2

■ "StorageWorks Optimum Operating Environment," page A-2

Physical and Electrical Specifications for the Controller

Table A–1 lists the physical and electrical specifications for the controller and cache modules.

Table A–1   Controller Specifications

Hardware	Length	Width	Power	Current at +5 V (nominal)	Current at +12 V (nominal)
HSZ70 Array Controller module	12.5 inches	8.75 inches	23.27 W	4.63 A	10 mA
Write-back Cache, 64 MB or 128 MB (Battery charging)	12.5 inches	7.75 inches	2.48 W 8.72 W	400 mA 400 mA	40 mA 560 mA

Environmental Specifications

The HSZ70 array controller is intended for installation in a Class A computer room environment. The environmental specifications listed in Table A–2, A—3, and A—4 are the same as for other Compaq storage devices.

Table A–2   StorageWorks Optimum Operating Environment

Condition	Description
Temperature	+18° to +24°C (+65° to +75°F)
Temperature rate of change	11°C (20°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 0.5 micron or larger, not to exceed 500,000 particles per cubic foot of air
Inlet air volume	0.026 cubic m per second (50 cubic ft per minute)

Table A–3   StorageWorks Maximum Operating Environment (Range)

Condition	Description
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/hour (20°F/hour) ±2°C/hour (4°F/hour)
Relative humidity	10% to 90% (noncondensing) Maximum wet bulb temperature: 28°C (82°F) Minimum dew point: 2°C (36°F)

Table A–4   StorageWorks Maximum Nonoperating Environment (Range)

Condition	Description
Temperature	-40° to +66°C (-40° to +151°F) (During transportation and associated short-term storage)
Relative Humidity	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 (MSL)

A P P E N D I X B

System Profiles

This appendix contains blank device and storageset profile sheets you can use to create your system profiles. It also contains enclosure templates you can use to help keep track of the location of devices and storagesets in your shelves:

Device Profile

TYPE

_____	Platter Disk Drive	_____	Optical Disk Drive
_____	Tape Drive	_____	CD-ROM

Device Name   _____

Unit Number   _____

Device Switches

Transportability

_____ No (default)

_____ Yes

Initialize Switches

Chunk Size	Save Configuration	Metadata
_____ Automatic (default)	_____ No (default)	_____ Destroy (default)
_____ 64 blocks	_____ Yes	_____ Retain
_____ 128 blocks		
_____ 256 blocks		
_____ Other:		

Unit Switches

Read Cache	Write Cache	Maximum Cache Transfer
_____ Yes(default)	_____ No (default)	_____ 32 Blocks (default)
_____ No	_____ Yes	_____ Other:

Availability	Write Protection
_____ Run (default)	_____ No (default)
_____ NoRun	_____ Yes

Device Profile

TYPE

_____ Platter Disk Drive   _____ Optical Disk Drive

_____ Tape Drive   _____ CD-ROM

Device Name   _____

Unit Number   _____

Device Switches

Transportability

_____ No (default)

_____ Yes

Initialize Switches

Chunk Size	Save Configuration	Metadata
_____ Automatic (default)	_____ No (default)	_____ Destroy (default)
_____ 64 blocks	_____ Yes	_____ Retain
_____ 128 blocks		
_____ 256 blocks		
_____ Other:		

Unit Switches

Read Cache	Write Cache	Maximum Cache Transfer
_____ Yes(default)	_____ No (default)	_____ 32 Blocks (default)
_____ No	_____ Yes	_____ Other:

Availability	Write Protection
_____ Run (default)	_____ No (default)
_____ NoRun	_____ Yes

Storageset Profile

Type of Storageset:

___ Mirrorset	___RAIDset	____ Stripeset	____ Striped Mirrorset	___JBOD

Storageset Name
Disk Drives
Unit Number

Partitions:

Unit #	Unit #	Unit #	Unit #	Unit #	Unit #	Unit #	Unit #
%	%	%	%	%	%	%	%

RAIDset Switches:

Reconstruction Policy	Reduced Membership	Replacement Policy
___Normal (default) ___Fast	___No (default) ___Yes, missing:	___Best performance (default) ___Best fit ___None

Mirrorset Switches:

Replacement Policy	Copy Policy	Read Source
___Best performance (default) ___Best fit ___None	___Normal (default) ___Fast	___Least busy (default) ___Round robin ___Disk drive:

Initialize Switches:

Chunk size	Save Configuration	Metadata
___ Automatic (default) ___ 64 blocks ___ 128 blocks ___ 256 blocks ___ Other:	___No (default) ___Yes	___Destroy (default) ___Retain

Unit Switches:

Read Cache	Write Cache	Maximum Cache Transfer
___ Yes (default) ___ No	___Yes (default) ___No	___32 blocks (default) ___Other:

Availability	Write Protection	
___Run (default) ___NoRun	___No (default) ___Yes	

Host Access Enabled:_____

Storageset Profile

Type of Storageset:

___ Mirrorset	___RAIDset	____ Stripeset	____ Striped Mirrorset	____JBOD

Storageset Name
Disk Drives
Unit Number

Partitions:

Unit #	Unit #	Unit #	Unit #	Unit #	Unit #	Unit #	Unit #
%	%	%	%	%	%	%	%

RAIDset Switches:

Reconstruction Policy	Reduced Membership	Replacement Policy
___Normal (default) ___Fast	___No (default) ___Yes, missing:	___Best performance (default) ___Best fit ___None

Mirrorset Switches:

Replacement Policy	Copy Policy	Read Source
___Best performance (default) ___Best fit ___None	___Normal (default) ___Fast	___Least busy (default) ___Round robin ___Disk drive:

Initialize Switches:

Chunk size	Save Configuration	Metadata
___ Automatic (default) ___ 64 blocks ___ 128 blocks ___ 256 blocks ___ Other:	___No (default) ___Yes	___Destroy (default) ___Retain

Unit Switches:

Read Cache	Write Cache	Maximum Cache Transfer
___ Yes (default) ___ No	___Yes (default) ___No	___32 blocks (default) ___Other:

Availability	Write Protection	
___Run (default) ___NoRun	___No (default) ___Yes	

Host Access Enabled:_____

Storage Map Template for PVA0

Use this template to:

■   Mark the location of your drives in the first enclosure in a multi-enclosure subsystem.

■   Mark the location of your drives in a enclosure subsystem.

Power Supply	Disk10300	Disk20300	Disk30300	Disk40300	Disk50300	Disk60300	Power Supply
Power Supply	Disk10200	Disk20200	Disk30200	Disk40200	Disk50200	Disk60200	Power Supply
Power Supply	Disk10100	Disk20100	Disk30100	Disk40100	Disk50100	Disk60100	Power Supply
Power Supply	Disk10000	Disk20000	Disk30000	Disk40000	Disk50000	Disk60000	Power Supply

Storage Map Template for PVA0

Use this template to:

■ Mark the location of your drives in the first enclosure in a multi-enclosure subsystem.

■ Mark the location of your drives in a enclosure subsystem.

Power Supply	Disk10300	Disk20300	Disk30300	Disk40300	Disk50300	Disk60300	Power Supply
Power Supply	Disk10200	Disk20200	Disk30200	Disk40200	Disk50200	Disk60200	Power Supply
Power Supply	Disk10100	Disk20100	Disk30100	Disk40100	Disk50100	Disk60100	Power Supply
Power Supply	Disk10000	Disk20000	Disk30000	Disk40000	Disk50000	Disk60000	Power Supply

Storage Map Template for PVA 2

Use this template to mark the location of your drives in the second enclosure in a multi-enclosure subsystem.

Power Supply	Disk11100	Disk21100	Disk31100	Disk41100	Disk51100	Disk61100	Power Supply
Power Supply	Disk11000	Disk21000	Disk31000	Disk41000	Disk51000	Disk61000	Power Supply
Power Supply	Disk10900	Disk20900	Disk30900	Disk40900	Disk50900	Disk60900	Power Supply
Power Supply	Disk10800	Disk20800	Disk30800	Disk40800	Disk50800	Disk60800	Power Supply

Storage Map Template for PVA 2

Use this template to mark the location of your drives in the second enclosure in a multi-enclosure subsystem.

Power Supply	Disk11100	Disk21100	Disk31100	Disk41100	Disk51100	Disk61100	Power Supply
Power Supply	Disk11000	Disk21000	Disk31000	Disk41000	Disk51000	Disk61000	Power Supply
Power Supply	Disk10900	Disk20900	Disk30900	Disk40900	Disk50900	Disk60900	Power Supply
Power Supply	Disk10800	Disk20800	Disk30800	Disk40800	Disk50800	Disk60800	Power Supply

Storage Map Template for PVA 3

Use this template to mark the location of your drives in the third enclosure in a multi-enclosure subsystem.

Power Supply	Disk11500	Disk21500	Disk31500	Disk41500	Disk51500	Disk61500	Power Supply
Power Supply	Disk11400	Disk21400	Disk31400	Disk41400	Disk51400	Disk61400	Power Supply
Power Supply	Disk11300	Disk21300	Disk31300	Disk41300	Disk51300	Disk61300	Power Supply
Power Supply	Disk11200	Disk21200	Disk31200	Disk41200	Disk51200	Disk61200	Power Supply

Storage Map Template for PVA 3

Use this template to mark the location of your drives in the third enclosure in a multi-enclosure subsystem.

Power Supply	Disk11500	Disk21500	Disk31500	Disk41500	Disk51500	Disk61500	Power Supply
Power Supply	Disk11400	Disk21400	Disk31400	Disk41400	Disk51400	Disk61400	Power Supply
Power Supply	Disk11300	Disk21300	Disk31300	Disk41300	Disk51300	Disk61300	Power Supply
Power Supply	Disk11200	Disk21200	Disk31200	Disk41200	Disk51200	Disk61200	Power Supply

Glossary

This glossary defines terms pertaining to the HSZ70 array controller. It is not a comprehensive glossary of computer terms.

adapter
A device that converts the protocol and hardware interface of one bus type into another without changing the function of the bus.

array controller
See controller.

autospare
A controller feature that automatically replaces a failed disk drive. To aid the controller in automatically replacing failed disk drives, you can enable the AUTOSPARE switch for the failedset causing physically replaced disk drives to be automatically placed into the spareset. Also called "autonewspare."

bad block
A data block that contains a physical defect.

backplane
The electronic printed circuit board into which you plug subsystem devices—for example, the SBB or power supply.

BBR
Bad Block Replacement. A replacement routine that substitutes defect-free disk blocks for those found to have defects. This process takes place in the controller, transparent to the host.

block
Also called a sector. The smallest collection of consecutive bytes addressable on a disk drive. In integrated storage elements, a block contains 512 bytes of data, error codes, flags, and the block's address header.

bootstrapping
A method used to bring a system or device into a defined state by means of its own action. For example, a machine routine whose first few instructions are enough to bring the rest of the routine into the computer from an input device.

cache memory
A portion of memory used to accelerate read and write operations.

CDU   Cable distribution unit. The power entry device for StorageWorks cabinets. The CDU provides the connections necessary to distribute power to the cabinet shelves and fans.

channel   Another term for a SCSI bus. *See also* SCSI.

chunk   A block of data written by the host.

chunk size   The number of data blocks, assigned by a system administrator, written to the primary RAIDset or stripeset member before the remaining data blocks are written to the next RAIDset or stripeset member.

CLI   Command line interpreter. The configuration interface to operate the controller software.

cold swap   A method of device replacement that requires the entire subsystem to be turned off before the device can be replaced. *See also* hot swap and warm swap.

configuration file   A file that contains a representation of a storage subsystem's configuration.

container   (1) Any entity that is capable of storing data, whether it is a physical device or a group of physical devices. (2) A virtual, internal, controller structure representing either a single disk or a group of disk drives linked as a storageset. Stripesets and mirrorsets are examples of storageset containers the controller uses to create units.

controller   A hardware device that, with proprietary software, facilitates communications between a host and one or more devices organized in an array. HS family controllers are examples of array controllers.

copying   A state in which data to be copied to the mirrorset is inconsistent with other members of the mirrorset. *See also* normalizing.

copying member   Any member that joins the mirrorset after the mirrorset is created is regarded as a copying member. Once all the data from the normal member (or members) is copied to a normalizing or copying member, the copying member then becomes a normal member. *See also* normalizing member.

data center cabinet   A generic reference to large DIGITAL subsystem cabinets, such as the SW600-series and 800-series cabinets in which StorageWorks components can be mounted.

data striping   The process of segmenting logically sequential data, such as a single file, so that segments can be written to multiple physical devices (usually disk drives) in a round-robin fashion. This technique is useful if the processor is capable of reading or writing data faster than a single disk can supply or accept the data. While data is being transferred from the first disk, the second disk can locate the next segment.

differential I/O module   A 16 bit I/O module with SCSI bus converter circuitry for extending a differential SCSI bus. *See also* I/O module.

differential SCSI bus   A bus in which a signal's level is determined by the potential difference between two wires. A differential bus is more robust and less subject to electrical noise than is a single-ended bus.

DILX   Disk inline exerciser. The controller's diagnostic software used to test the data transfer capabilities of disk drives in a way that simulates a high level of user activity.

dirty data   The write-back cached data that has not been written to storage media, even though the host operation processing the data has completed.

DOC   DWZZA-On-a-Chip. An NCR53C120 SCSI bus extender chip used to connect a SCSI bus in an expansion cabinet to the corresponding SCSI bus in another cabinet.

dual-redundant configuration   A controller configuration consisting of two active controllers operating as a single controller. If one controller fails, the other controller assumes control of the failing controller's devices.

DUART   Dual universal asynchronous receiver and transmitter. An integrated circuit containing two serial, asynchronous transceiver circuits.

DWZZA   A StorageWorks SCSI-bus-signal converter used to connect 8-bit single-ended devices to hosts with 16-bit differential SCSI adapters. This converter extends the range of a single-ended SCSI cable to the limit of a differential SCSI cable. *See also* SCSI bus signal converter.

DWZZB   A StorageWorks SCSI bus signal converter used to connect a variety of 16-bit single-ended devices to hosts with 16-bit differential SCSI adapters. *See also* SCSI bus signal converter.

DWZZC   The 16-bit SCSI table-top SCSI bus signal converter used to extend a differential SCSI bus, or connect a differential SCSI bus to a single ended SCSI bus. *See also* SCSI bus signal converter.

ECB   External cache battery. The unit that supplies backup power to the cache module in the event the primary power source fails or is interrupted.

EMU   Environmental monitoring unit. A unit that provides increased protection against catastrophic failures. Some subsystem enclosures include an EMU which works with the controller to detect conditions such as failed power supplies, failed blowers, elevated temperatures, and external air sense faults. The EMU also controls certain cabinet hardware including DOC chips, alarms, and fan speeds.

ESD   Electrostatic discharge. The discharge of potentially harmful static electrical voltage as a result of improper grounding.

extended subsystem   A subsystem in which two cabinets are connected to the primary cabinet.

external cache battery   *See* ECB.

Failback   The process of restoring data access to the newly-restored controller in a dual-redundant controller configuration (see Failover).

failedset   A group of failed mirrorset or RAIDset devices automatically created by the controller.

failover   The process that takes place when one controller in a dual-redundant configuration assumes the workload of a failed companion controller. Failover continues until the failed controller is repaired or replaced.

FCC   Federal Communications Commission. The federal agency responsible for establishing standards and approving electronic devices within the United States.

FCC Class A   This certification label appears on electronic devices that can only be used in a commercial environment within the United States.

FCC Class B   This certification label appears on electronic devices that can be used in either a home or a commercial environment within the United States.

FD SCSI   The fast, narrow, differential SCSI bus with an 8-bit data transfer rate of 10 MB/s. *See also* FWD SCSI and SCSI.

flush   The act of writing dirty data from cache to a storage media.

forced errors   A data bit indicating a corresponding logical data block contains unrecoverable data.

FRU   Field replaceable unit. A hardware component that can be replaced at the customer's location by DIGITAL service personnel or qualified customer service personnel.

FWD SCSI   A fast, wide, differential SCSI bus with a maximum 16-bit data transfer rate of 20 MB/s. *See also* SCSI and FD SCSI.

host   The primary or controlling computer to which a storage subsystem is attached.

host adapter   A device that connects a host system to a SCSI bus. The host adapter usually performs the lowest layers of the SCSI protocol. This function may be logically and physically integrated into the host system.

host compatibility mode   A setting used by the controller to provide optimal controller performance with specific operating systems. This improves the controller's performance and compatibility with the specified operating system. The supported modes are: A, Normal (including DIGITAL UNIX®, OpenVMS, Sun®, and Hewlett-Packard® HP–UX); B, IBM AIX®; C, Proprietary; and D, Microsoft Windows NT™ Server.

hot disks   A disk containing multiple hot spots. Hot disks occur when the workload is poorly distributed across storage devices and prevents optimum subsystem performance. *See also* hot spots.

hot spots   A portion of a disk drive frequently accessed by the host. Because the data being accessed is concentrated in one area, rather than spread

across an array of disks providing parallel access, I/O performance is significantly reduced. *See als*o hot disks.

hot swap A method of device replacement that allows normal I/O activity on a device's bus to remain active during device removal and insertion. The device being removed or inserted is the only device that cannot perform operations during this process. *See also* cold swap and warm swap.

HSOF Hierarchical Storage Operating Firmware. Software contained on a removable ROM program card that provides the operating system for the array controller.

initiator A SCSI device that requests an I/O process to be performed by another SCSI device, namely, the SCSI target. The controller is the initiator on the device bus. The host is the initiator on the host bus.

instance code A four-byte value displayed in most text error messages and issued by the controller when a subsystem error occurs. The instance code indicates when during software processing the error was detected.

I/O module A 16-bit SBB shelf device that integrates the SBB shelf with either an 8-bit single ended, 16-bit single-ended, or 16-bit differential SCSI bus. *See also* I/O Module.

JBOD Just a bunch of disks. A term used to describe a group of single-device logical units.

local connection A connection to the subsystem using either its serial maintenance port or the host's SCSI bus. A local connection enables you to connect to one subsystem controller within the physical range of the serial or host SCSI cable.

local terminal A terminal plugged into the EIA-423 maintenance port located on the front bezel of the controller. *See also* maintenance terminal.

logical bus A single-ended bus connected to a differential bus by a SCSI bus signal converter.

logical unit A physical or virtual device addressable through a target ID number. LUNs use their target's bus connection to communicate on the SCSI bus.

logical unit number	A value that identifies a specific logical unit belonging to a SCSI target ID number. A number associated with a physical device unit during a task's I/O operations. Each task in the system must establish its own correspondence between logical unit numbers and physical devices.
LRU	Least recently used. A cache term used to describe the block replacement policy for read cache.
maintenance terminal	An EIA-423-compatible terminal used with the controller. This terminal is used to identify the controller, enable host paths, enter configuration information, and check the controller's status. The maintenance terminal is not required for normal operations. *See also* local terminal.
MB/s	Megabytes per second. A unit designation used to measure the SCSI bus bandwidth.
member	A container that is a storage element in a RAID array.
metadata	The data written to a disk for the purposes of controller administration. Metadata improves error detection and media defect management for the disk drive. It is also used to support storageset configuration and partitioning. Non-transportable disks also contain metadata to indicate they are uniquely configured for StorageWorks environments. Metadata can be thought of as "data about data."
mirroring	The act of creating an exact copy or image of data.
mirrored write-back caching	A method of caching data that maintains two copies of the cached data. The copy is available if either cache module fails.
mirrorset	*See* RAID level 1.
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" membership.
non-redundant controller configuration	(1) A single controller configuration. (2) A controller configuration which does not include a second controller.

normal member	A mirrorset member that, block-for-block, contains the same data as other normal members within the mirrorset. Read requests from the host are always satisfied by normal members.
normalizing	Normalizing is a state in which, block-for-block, data written by the host to a mirrorset member is consistent with the data on other normal and normalizing members. The normalizing state exists only after a mirrorset is initialized. Therefore, no customer data is on the mirrorset.
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 lost cache data was cleared. A normalizing member is created by a normal member when either all of the normal members fail or all of the normal members are removed from the mirrorset. *See also* copying member.
NVM	Non-Volatile Memory. A type of memory—the contents of which survive loss of power.
OCP	Operator control panel. The control or indicator panel associated with a device. The OCP is usually mounted on the device and is accessible to the operator.
other controller	The controller in a dual-redundant pair that's connected to the controller that's serving your current CLI session. *See also* this controller.
PCM	Polycenter Console Manager.
PCMCIA	Personal Computer Memory Card Industry Association. An international association formed to promote a common standard for PC card-based peripherals to be plugged into notebook computers. The card commonly known as a PCMCIA card is about the size of a credit card.
parity	A method of checking if binary numbers or characters are correct by counting the ONE bits. In odd parity, the total number of ONE bits must be odd; in even parity, the total number of ONE bits must be even. Parity information can be used to correct corrupted data. RAIDsets use parity to improve the availability of data.

parity bit	A binary digit added to a group of bits that checks to see if there are errors in the transmission.
parity RAID	*See* RAIDset.
partition	A logical division of a container, represented to the host as a logical unit.
port	(1) In general terms, a logical channel in a communications system. (2) The hardware and software used to connect a host controller to a communications bus, such as a SCSI bus or serial bus. Regarding the controller, the port is: (1) the logical route for data in and out of a controller that can contain one or more channels, all of which contain the same type of data. (2) The hardware and software that connects a controller to a SCSI device.
primary cabinet	The primary cabinet is the subsystem enclosure that contains the controllers, cache modules, external cache batteries, and the PVA module.
program card	The PCMCIA card containing the controller's operating software.
PTL	Port-Target-LUN. The controller's method of locating a device on the controller's device bus.
PVA module	Power Verification and Addressing module.
quiesce	The act of rendering bus activity inactive or dormant. For example, "quiesce the SCSI bus operations during a device warm-swap."
RAID	Redundant Array of Independent Disks. Represents multiple levels of storage access developed to improve performance or availability or both.
RAID level 0	A RAID storageset that stripes data across an array of disk drives. A single logical disk spans multiple physical disks, allowing parallel data processing for increased I/O performance. While the performance characteristics of RAID level 0 is excellent, this is the only RAID level that does not provide redundancy. Raid level 0 storagesets are sometimes referred to as stripesets.

RAID level 0+1   A RAID storageset that stripes data across an array of disks (RAID level 0) and mirrors the striped data (RAID level 1) to provide high I/O performance and high availability. This RAID level is alternatively called a striped mirrorset. Raid level 0+1 storagesets are sometimes referred to as striped mirrorsets.

RAID level 1   A RAID storageset of two or more physical disks that maintains a complete and independent copy of the entire virtual disk's data. This type of storageset has the advantage of being highly reliable and extremely tolerant of device failure. Raid level 1 storagesets are sometimes referred to as mirrorsets.

RAID level 3   A RAID storageset that transfers data parallel across the array's disk drives a byte at a time, causing individual blocks of data to be spread over several disks serving as one enormous virtual disk. A separate redundant check disk for the entire array stores parity on a dedicated disk drive within the storageset. *Contrast* RAID level 5.

RAID Level 5   A RAID storageset that, unlike RAID level 3, stores the parity information across all of the disk drives within the storageset. *Contrast* RAID level 3.

RAID level 3/5   A DIGITAL-developed RAID storageset that stripes data and parity across three or more members in a disk array. A RAIDset combines the best characteristics of RAID level 3 and RAID level 5. A RAIDset is the best choice for most applications with small to medium I/O requests, unless the application is write intensive. A RAIDset is sometimes called parity RAID. Raid level 3/5 storagesets are sometimes referred to as RAIDsets.

RAIDset   *See* RAID level 3/5.

read caching   A cache management method used to decrease the subsystem's response time to a read request by allowing the controller to satisfy the request from the cache memory rather than from the disk drives.

reconstruction   The process of regenerating the contents of a failed member's data. The reconstruct process writes the data to a spareset disk and then incorporates the spareset disk into the mirrorset, striped mirrorset, or RAIDset from which the failed member came. *See also* regeneration.

reduced   Indicates that a mirrorset or RAIDset is missing one member because the member has failed or has been physically removed.

redundancy   The provision of multiple interchangeable components to perform a single function in order to cope with failures and errors. A RAIDset is considered to be redundant when user data is recorded directly to one member and all of the other members include associated parity information.

regeneration   (1) The process of calculating missing data from redundant data. (2) The process of recreating a portion of the data from a failing or failed drive using the data and parity information from the other members within the storageset. The regeneration of an entire RAIDset member is called reconstruction. *See also* reconstruction.

RFI   Radio frequency interference. The disturbance of a signal by an unwanted radio signal or frequency.

replacement policy The policy specified by a switch with the SET FAILEDSET command indicating whether a failed disk from a mirrorset or RAIDset is to be automatically replaced with a disk from the spareset. The two switch choices are AUTOSPARE and NOAUTOSPARE.

SBB   StorageWorks building block. (1) A modular carrier plus the interface required to mount the carrier into a standard StorageWorks shelf. (2) any device conforming to shelf mechanical and electrical standards installed in a 3.5-inch or 5.25-inch carrier, whether it is a storage device or power supply.

SCSI   Small computer system interface. (1) An ANSI interface standard defining the physical and electrical parameters of a parallel I/O bus used to connect initiators to devices. (2) a processor-independent standard protocol for system-level interfacing between a computer and intelligent devices including hard drives, floppy disks, CD-ROMs, printers, scanners, and others.

SCSI-A cable   A 50-conductor (25 twisted-pair) cable generally used for single-ended, SCSI-bus connections.

SCSI bus signal converter Sometimes referred to as an adapter. (1) A device used to interface between the subsystem and a peripheral device unable to be mounted directly into the SBB shelf of the subsystem. (2) a device used to

	connect a differential SCSI bus to a single-ended SCSI bus. (3) A device used to extend the length of a differential or single-ended SCSI bus. *See also* DWZZA, DWZZB, DWZZC, and I/O module.
SCSI device	(1) A host computer adapter, a peripheral controller, or an intelligent peripheral that can be attached to the SCSI bus. (2) Any physical unit that can communicate on a SCSI bus.
SCSI device ID number	A bit-significant representation of the SCSI address referring to one of the signal lines, numbered 0 through 7 for an 8-bit bus, or 0 through 15 for a 16-bit bus. *See also* target ID number.
SCSI ID number	The representation of the SCSI address that refers to one of the signal lines numbered 0 through 15.
SCSI-P cable	A 68-conductor (34 twisted-pair) cable generally used for differential bus connections.
SCSI port	(1) Software: The channel controlling communications to and from a specific SCSI bus in the system. (2) Hardware: The name of the logical socket at the back of the system unit to which a SCSI device is connected.
signal converter	*See* SCSI bus signal converter.
single ended I/O module	A 16-bit I/O module. *See also* I/O module.
single-ended SCSI bus	An electrical connection where one wire carries the signal and another wire or shield is connected to electrical ground. Each signal's logic level is determined by the voltage of a single wire in relation to ground. This is in contrast to a differential connection where the second wire carries an inverted signal.
spareset	A collection of disk drives made ready by the controller to replace failed members of a storageset.
storage array	An integrated set of storage devices.
storage array subsystem	*See* storage subsystem.

storageset	(1) A group of devices configured with RAID techniques to operate as a single container. (2) Any collection of containers, such as stripesets, mirrorsets, striped mirrorsets, and RAIDsets.
storage subsystem	The controllers, storage devices, shelves, cables, and power supplies used to form a mass storage subsystem.
storage unit	The general term that refers to storagesets, single-disk units, and all other storage devices that are installed in your subsystem and accessed by the host. A storage unit can be any entity that is capable of storing data, whether it is a physical device or a group of physical devices.
StorageWorks	A family of DIGITAL modular data storage products which allow 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 systems include integrated SBBs and array controllers to form storage subsystems. System-level enclosures to house the shelves and standard mounting devices for SBBs are also included.
stripe	The data divided into blocks and written across two or more member disks in an array.
striped mirrorset	See RAID level 0+1.
stripeset	*See* RAID level 0.
stripe size	The stripe capacity as determined by $n{-}1$ times the chunksize, where n is the number of RAIDset members.
striping	The technique used to divide data into segments, also called chunks. The segments are striped, or distributed, across members of the stripeset. This technique helps to distribute hot spots across the array of physical devices to prevent hot spots and hot disks. Each stripeset member receives an equal share of the I/O request load, improving performance.
surviving controller	The controller in a dual-redundant configuration pair that serves its companion's devices when the companion controller fails.

tape   A storage device supporting sequential access to variable sized data
                                records.

Tape Inline Exerciser   *See* TILX.

target   (1) A SCSI device that performs an operation requested by an initiator.
                                (2) Designates the target identification (ID) number of the device.

target ID number   The address a bus initiator uses to connect with a bus target. Each bus
                                target is assigned a unique target address.

this controller   The controller that is serving your current CLI session through a local
                                or remote terminal. *See also* other controller.

TILX   Tape inline exerciser. The controller's diagnostic software to test the
                                data transfer capabilities of tape drives in a way that simulates a high
                                level of user activity.

Ultra SCSI bus   A wide, fast-20 SCSI bus.

unit   A container made accessible to a host. A unit may be created from a
                                single disk drive or tape drive. A unit may also be created from a more
                                complex container such as a RAIDset. The controller supports a
                                maximum of eight units on each target. *See also* target and target ID
                                number.

unwritten cached data   Sometimes called unflushed data. *See* dirty data.

UPS   Uninterruptible power supply. A battery-powered power supply
                                guaranteed to provide power to an electrical device in the event of an
                                unexpected interruption to the primary power supply. Uninterruptible
                                power supplies are usually rated by the amount of voltage supplied and
                                the length of time the voltage is supplied.

VHDCI   Very high-density-cable interface. A 68-pin interface. Required for
                                ultra-SCSI connections.

virtual terminal   A software path from an operator terminal on the host to the controller's
                                CLI interface, sometimes called a host console. The path can be
                                established via the host port on the controller (using HSZ term) or via
                                the maintenance port through an intermediary host.

warm swap	A device replacement method that allows the complete system remains online during device removal or insertion. The system bus may be halted, or quiesced, for a brief period of time during the warm-swap procedure.
write-back caching	A cache management method used to decrease the subsystem's response time to write requests by allowing the controller to declare the write operation "complete" as soon as the data reaches its cache memory. The controller performs the slower operation of writing the data to the disk drives at a later time.
write-through caching	A cache management method used to decrease the subsystem's response time to a read. This method allows the controller to satisfy the request from the cache memory rather than from the disk drives.
write hole	The period of time in a RAID level 1 or RAID level 5 write operation when there is an opportunity for undetectable RAIDset data corruption. Write holes occur under conditions such as power outages, where the writing of multiple members can be abruptly interrupted. A battery backed-up cache design eliminates the write hole because data is preserved in cache and unsuccessful write operations can be retried.
write-through cache	A cache management technique for retaining host write requests in read cache. When the host requests a write operation, the controller writes data directly to the storage device. This technique allows the controller to complete some read requests from the cache, greatly improving the response time to retrieve data. The operation is complete only after the data to be written is received by the target storage device. This cache management method may update, invalidate, or delete data from the cache memory accordingly, to ensure that the cache contains the most current data.

Index

D

W